Near-Regular Structure Discovery Using Linear Programming

Qixing Huang
Stanford University
and
Leonidas J. Guibas
Stanford University
and
Niloy J. Mitra
University College London

Near-regular structures are common in manmade and natural objects. Algorithmic detection of such regularity greatly facilitates our understanding of shape structures, leads to compact encoding of input geometries, and enables efficient generation and manipulation of complex patterns on both acquired and synthesized objects. Such regularity manifests itself both in the repetition of certain geometric elements, as well as in the structured arrangement of the elements. We cast the regularity detection problem as an optimization and efficiently solve it using linear programming techniques. Our optimization has a discrete aspect, i.e., the connectivity relationships among the elements; as well as a continuous aspect, i.e., the locations of the elements of interest. Both these aspects are captured by our near-regular structure extraction framework, which alternates between discrete and continuous optimizations. We demonstrate the effectiveness of our framework on a variety of problems including near-regular structure extraction, structure-preserving pattern manipulation, and markerless correspondence detection. Robustness results with respect to geometric and topological noise are presented on synthesized, realworld, and also benchmark datasets.

Categories and Subject Descriptors: I.3.5 [**Computer Graphics**]: Computational Geometry and Object Modeling; I.3.6 [**Computer Graphics**]: Methodology and Techniques

Additional Key Words and Phrases: intrinsic near-regular structure, integer and linear programming, pattern manipulation, markerless correspondence

1. INTRODUCTION

Global structures in the form of symmetric and near-regular repeating patterns are common in natural and manmade objects (see Figure 1). Such structures arise both from the presence of regularly spaced repeated elements, as well as from geometric and topological consistency in the arrangement of the elements. Discovering such global spatial arrangements entails a holistic understanding of object geometry that goes well beyond the detection of localized features [Thompson 1945]. Moreover, regularity is often *partial* involving only parts of an object; and *approximate*, in that the consistency of the repeated elements and of their spatial relationships may be imprecise. These make computational discovery of near-regularity particularly challenging. In recent years, the problem of regularity detection of 3D geometry under rigid and isometric mappings has attracted a great deal of attention both because of its fundamental importance in shape understanding as well as its applications in shape matching, structure-preserving editing, and structure-driven shape synthesis [Pauly et al. 2008; Lipman and Funkhouser 2009; Bokeloh et al. 2010; Kim et al. 2010; Mitra et al. 2010; Mitra et al. 2012]. In this paper, we introduce a novel framework to discover such near-regular structures, both in terms of the representation of the regularity and in terms of an efficient extraction algorithm to facilitate a variety of application (see Figure 2).

Our goal is to discover near-regular repeated structures on objects. In this paper, we treat objects as surfaces, while ignoring internal structure. We assume such surfaces to be represented as piecewise linearly embedded meshes with the topology of a 2-manifold. Near-regular repeated structures manifest as repeated surface elements that are regularly arranged on the surface. We encode and extract such structures using a mesh-like structure $\mathcal{M}$ called near-regular geodesic subdivision or *NRG-subdivision* (see Figure 3). The 0-dimensional elements of $\mathcal{M}$, which we call *location points*, are points on the surface that capture the locations of the repeating elements of the structure; the 1-dimensional elements of $\mathcal{M}$, which we call *curves*, are the shortest geodesic curves on the surface that connect pairs of adjacent locations points; and the 2-dimensional elements of $\mathcal{M}$, which we call *patches*, are surface segments that are enclosed by loops of these shortest geodesic curves. Note that such a patch is not necessarily a topological disk, which is a key difference between our mesh-like structure and a standard polygonal mesh. We allow this relaxation because in many practical applications topological noise resulting from surface scans would otherwise prevent us from discovering the geometric regularity of a shape. In Section 3, we formally define the objects that constitute such a NRG-subdivision.

The central task in near-regular structure extraction is to determine a NRG-subdivision $\mathcal{M}$ that captures the locations of the repeating elements. Once their locations (i.e., the 0-dimensional entities of $\mathcal{M}$) are determined, the repeating elements, which we call

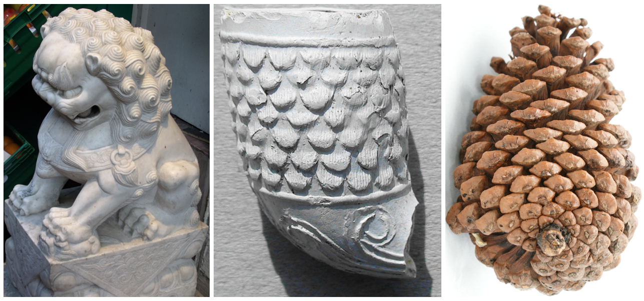


Fig. 1: *Near-regular structures, i.e., regularly arranged near-repeated geometric elements, are commonly found in many natural and manmade objects. Automatically discovering such structures, i.e., which elements are repeated and how they are spatially arranged, is challenging.*

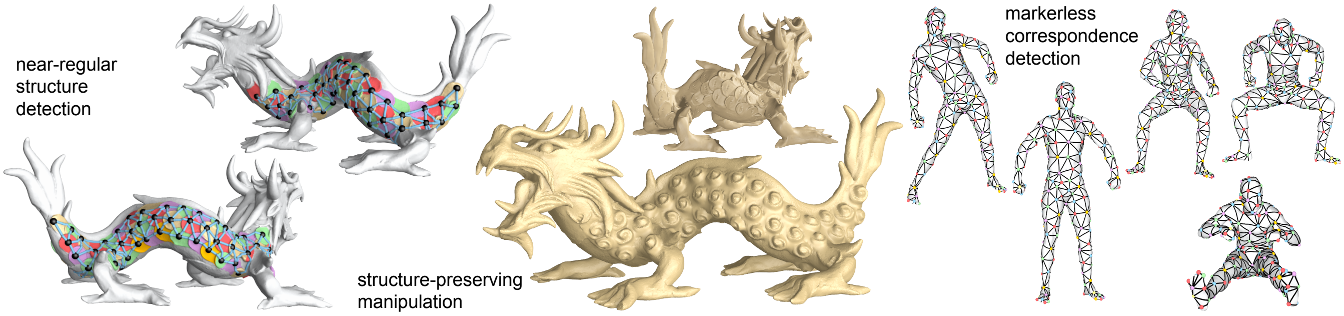


Fig. 2: *Various applications enabled by our near-regular structure extraction framework.*

texels, can be extracted by solving a variational problem (see Section 4.1). Note that if two texels χ_i and χ_j are adjacent, the corresponding 0-dimensional location points p_i and p_j representing these texels must have a unique shortest curve (or geodesic) between them. We encode such a mesh-like structure as $\mathcal{M}$.

We extract $\mathcal{M}$ by solving a constrained optimization with both geometric and topological aspects, translating into a mixed discrete-continuous problem involving appropriate *regularity measures*. On the geometry side, we compute candidate location points as points with consistent feature descriptors. On the topology side, we build either a triangular or a quadrilateral subdivision of the surface using the shortest geodesic curves connecting a subset of the candidate location points. Our formulation is in the same spirit as the method proposed by Hays et al. [2006] for detecting texture regularities in images, though the technical details, particularly on the topology side, are completely different.

Our method for extracting $\mathcal{M}$ takes as input a connected, oriented, piecewise linear 2-manifold, represented as a triangle mesh in Euclidean space; a set of points in the mesh that we call *sample points*, which is sufficiently large (see later); and a conservative estimate of the maximum geodesic distance between adjacent sample points. (Note that we assume there is a unique shortest geodesic between any two sufficiently close (see later) sample points.) We then enumerate all possible patches using the sample points as vertices and the shortest geodesic curves as patch boundaries. Extracting $\mathcal{M}$ then amounts to selecting the subset of patches that *maximizes* a suitable regularity measure, while forming a geodesic subdivision of the input surface. We achieve this by finding a consistent assignment of indicator variables, capturing selection among the set of candidate patches, using a sparse integer program. We solve the integer program through a tight linear relaxation, thereby establishing topological connectivity, while automatically enforcing regularity of the extracted geometric subdivision. Note that we benefit from recent advances in solving large scale sparse linear programs that allow us to robustly and efficiently solve systems involving 50-100k variables in the order of seconds [Grant and Boyd 2011]. We also provide corresponding integrality conditions characterizing the uniqueness of the proposed relaxation strategy. Furthermore, a simple modification of the above approach allows detection of *partial* NRG-subdivisions covering only part of the input surface.

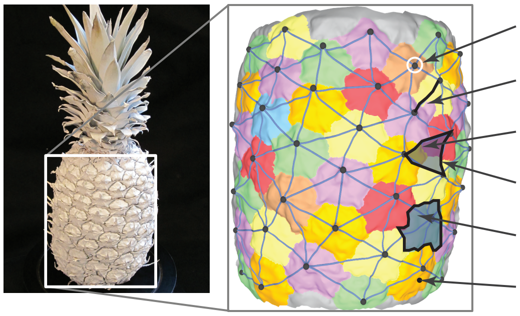


Fig. 3: *Near-regular structures detected on a scanned pineapple (see also Figure 4). Starting from the input surface represented as a triangle mesh $\mathcal{S}$, we obtain a set of sample points (only a few shown). The extracted NRG-subdivision is encoded as a mesh-like structure $\mathcal{M}$, where the 0-dimensional entities are called the location points (a subset of the sample points); the 1-dimensional entities are the shortest geodesic curves connecting neighboring location points; and the 2-dimensional entities are called the patches. The patch boundaries, which we call loops, are formed by strings of geodesic curves. Near-regularity is estimated by measuring regularity of the topological connectivity of NRG-subdivision $\mathcal{M}$ and an application specific regularity of the geometric patches. The extracted NRG-subdivision $\mathcal{M}$ can then be used to further extract the repeated elements, which we call texels, as certain surface segments around the location points.*

We apply the extraction method outlined above as follows: First, apply the method to detect near-regular structures on input surfaces to extract near-regular repeating elements. Starting from an initial set of sample points with consistent feature descriptors, we build a NRG-subdivision, use the currently detected structure to refine the set of sample points, and extract an updated NRG-subdivision. We iterate this procedure until the structure stabilizes, and then compute the repeating elements using a variational formulation. We also show applications to structure-aware shape editing. Second, we extend the framework to establish a consistent structure between two shapes, i.e., address the markerless correspondence problem. In this scenario, the sample points, shortest geodesic curves, and patches in the standard setting become point correspondences, curve correspondences, and patch correspondences, respectively. The extracted general NRG-subdivision represents a match between a pair of extracted structures from both shapes. Experimental results show that our method compares favorably against state-of-the-art alternatives.

In summary, our main contributions are:

—a new formulation of the regularity discovery as the detection of a partial NRG-subdivision on the underlying surface posed as an integer programming problem, and solved using a tight and efficient linear relaxation strategy with performance guarantees;

—a practical method with alternating topological and geometric optimizations for extracting NRG-subdivision with only weak restrictions on the initial sample placements; and

—applications of the framework for near-regular structure detection, structure-preserving pattern replacement on surfaces, and markerless correspondence extraction between shape pairs.

2. RELATED WORK

Detection of repetitions and patterns in geometric data, motivated by a desire for better understanding and encoding of natural and man-made objects, remains an important topic in shape analysis since early works on allometry aimed at exploring relationship between size and shape of forms [Thompson 1945]. In this section, we briefly discuss the relevant advances in this direction.

Shape analysis. In the case of texture and image analysis, Schaffalitzky and Zissermann [1999] perform edge detection to discover candidate elements and greedily grow the patterns to translational grids based on maximum likelihood estimation; Tuytelaars et al. [2003] use a Hough transform based hashing scheme to detect regular patterns in images under perspective skew; while Liu et al. [2004] propose a user-seeded lattice extraction framework for analyzing commonly occurring near-regular textures. Hays et al. [2006] detect lattices of near-regular textures by alternating between a geometric step, which optimizes locations of features and a topological step, which extracts near-regular textures from optimized features. In particular, the topological step is formulated as a higher order correspondence problem and is solved using a spectral approach. Our method is conceptually similar to their method, although the specifics of both steps are very different. Park et al. [2009] propose a novel formulation for robust detection of 2D lattices based on a Markov random field on candidate lattice basis vectors obtained using point-based voting coupled with a thin-plate spline deformation. These methods are difficult to extend for handling 3D geometry, where a canonical parameterization is typically lacking, and data is often corrupted with noise, outliers, or can be incomplete due to occlusion artifacts.

In the context of extrinsic and intrinsic symmetry detection, various techniques have been proposed including transform domain voting [Mitra et al. 2006], Monte Carlo sampling-based estimation of continuous symmetry transform [Podolak et al. 2006], eigenanalysis of the Laplace-Beltrami operator for global intrinsic symmetry [Ovsjanikov et al. 2008], and voting based partial intrinsic symmetry extraction [Xu et al. 2009]. Our goal is to go beyond detecting repeated patterns across texel pairs and reveal the regularity in how the texels are laid out with respect to each other.

In terms of regularity detection, Langbein and Martin [2006] propose regularity feature trees as a concise description of symmetry features for capturing important geometric design intent of CAD models, while Liu et al. [2007] rely on user annotations to initiate segmentation of triangle meshes into periodic reliefs. Pauly et al. [2008] propose a computational framework for discovering regular or repeated structures in 3D geometries using a non-linear optimization to search for characteristic regularity grids, involving 1-, 2-, or 3-parameters, in an appropriate transform domain. The approach has subsequently been generalized to detect intrinsic regularity on developable surfaces [Mitra et al. 2010]. However, in cases when the repeated patterns are small and the repetitions are only approximate, regularities are hard to detect using transform domain clustering, due to the lack of a reliable and stable initialization. In contrast, since we simultaneously maximize for patch regularity and connectivity regularity in the topology reconstruction phase, we can extract NRG-subdivision even when individually the regularity or geometric repetitions are low, e.g., the small scales of the dragon as in Figure 2. Note that previous methods fail to detect NRG-subdivision on meshes as shown in Figures 7, 6, 8, and 15. Patches, along with their repetition patterns as extracted using our framework, can be directly used as inputs for intuitive manipulation [Gal et al. 2009; Bokeloh et al. 2010], and for exploiting subspace symmetries [Berner et al. 2011] (see also survey [Mitra et al. 2012]).

Correspondence detection. Establishing a meaningful correspondence map between a pair of shapes is one of the central problems in geometry processing. When the models start in arbitrary poses, the problem is challenging given the exponential size of the solution space (see survey [van Kaick et al. 2011]). In the context of isometric mappings, where pairwise geodesic distances are preserved, Angulelov et al. [2005] perform markerless registration of mesh pairs using a joint probabilistic model over all point-to-point correspondences assignments, seeded with spin-image based descriptors. Lipman and Funkhouser [2009] observe that isometric transforms are contained in the family of conformal mappings, and hence conformally map shapes to the complex plane via uniformization, and iteratively use possible assignments of point triplets to compute aligning Möbius transforms. The candidate transforms are then used to vote for correspondences for the other points. Subsequently, the method has been extended to detect global intrinsic symmetry [Kim et al. 2010]. For surfaces undergoing approximate isometric deformations, Tevs et al. [2009] propose a RANSAC-based correspondence detection algorithm using geodesic distance checking to verify isometric equivalence. Recently, Ovsjanikov et al. [2010] study the structure of the space of isometric correspondences, and under mild conditions present the surprising result that a single correspondence is sufficient to recover isometries for entire shapes. We treat the isometric correspondence detection problem in our NRG-subdivision extraction framework, and demonstrate comparable performance with specialized correspondence detection algorithms, even under moderate deviations and non-trivial topological variations.

3. SAMPLE-BASED NRG-SUBDIVISION EXTRACTION

In this section, we introduce *sample-based NRG-subdivision extraction*, i.e., extracting a geodesic near-regular subdivision from a candidate set of sample points on the input model. We assume that we are given a compact, connected, oriented, triangulated surface $\mathcal{S}$, possibly with boundary, that is piecewise linearly embedded in Euclidean space. We assume that $\mathcal{S}$ is represented as a triangular mesh $(\mathcal{V}, \mathcal{E}, \mathcal{F})$ where $\mathcal{V}$, $\mathcal{E}$, and $\mathcal{F}$ denote the sets of vertices, edges, and triangular faces of $\mathcal{S}$, respectively. (We will abuse notation slightly and let $\mathcal{S}$ denote both the mesh and the underlying surface.)

Figure 4 shows the pipeline of the proposed sample-based NRG-subdivision extraction. The input consists of a set of sample points $\mathcal{V}_s \subset \mathcal{V}$, a conservative estimate $d_{\max}$ of the maximum length of (the shortest) geodesic distances between pairs of adjacent points of the NRG-subdivision $\mathcal{M}$ to be extracted, and for each vertex in $\mathcal{M}$, a canonical number η of adjacent patches (e.g., $\eta = 4$ for quadrilateral patches and $\eta = 6$ for triangular patches). Sample-based NRG-subdivision extraction now proceeds in two stages: First, we generate a set of candidate patches by enumerating all possible loops of geodesic curves, where each geodesic curve connects a pair of sample points and its length is below d_{max}. Second, we seek a subset of the candidate patches among those created in first step such that (i) the candidate patches form a valid patch collection or as we

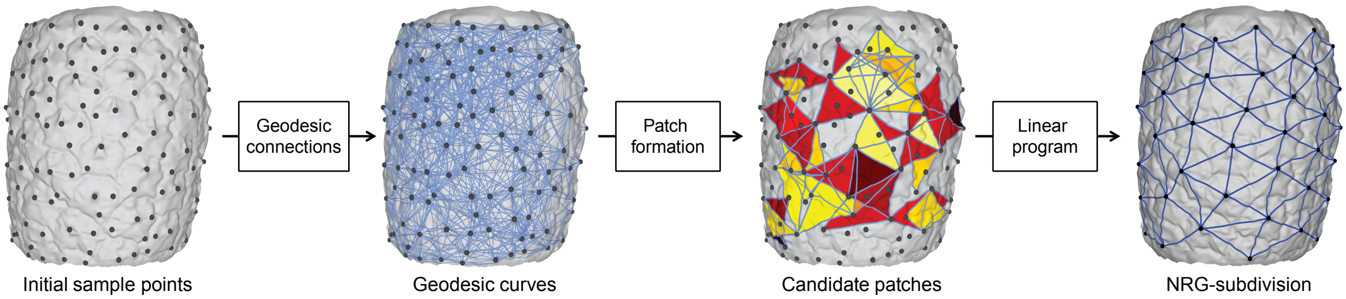


Fig. 4: *The pipeline of sample-based NRG-subdivision extraction. Starting from a set of sample points on the input oriented manifold, we generate a set of overlapping candidate patches by connecting neighboring sample points with the shortest geodesic curves. NRG-subdivision extraction then amounts to selecting a subset of patches to maximize an appropriately defined regularity score, subject to the subset of patches forming a valid geodesic subvision.*

call a *geodesic subdivision*, and (ii) the geodesic subdivision with the highest regularity score is selected.

The rest of this section is organized as follows: In Section 3.1, we describe how to compute candidate patches on triangular meshes; in Section 3.2, we define geodesic subdivisions and present necessary and sufficient conditions on a subset of candidate patches to form a geodesic subdivision; in Section 3.3, we show how to formulate NRG-subdivision extraction as solving an integer program; and finally in Section 3.4, we describe how to solve the induced integer program by solving its linear programming relaxation.

3.1 Candidate Patch Generation and Representation

Given the set of sample points $\mathcal{V}_s$, our goal at this stage is to compute a large collection of candidate patches bounded by curves connecting points from $\mathcal{V}_s$. We achieve this in four steps: (a) compute candidate curves; (b) assemble these curves into valid loops; (c) perturb the loops to simplify their representation; and (d) select as candidate patches the interiors of valid perturbed loops.

(a) Compute candidate curves. To begin, we compute a candidate curve set $\mathcal{C}$ consisting of one of two kinds of curves each having length less than $d_{\max}$ between pairs of points as follows. Given a pair of sample points v and v' in $\mathcal{V}_s$, two situations arise: Either, (i) v and v' do not both belong to the same connected component of the mesh boundary $\partial\mathcal{S}$ and we connect them by the shortest geodesic curve between them using the method introduced in Xin et al. [2009]. Note that the resulting geodesic curve is a polygonal curve that may cut through faces of $\mathcal{S}$. We refer to such curves as *interior curves*. Or, (ii) we connect v and v' by the shorter path along the (closed) boundary $\partial\mathcal{S}$, arbitrarily breaking ties when the two paths are of the same length. We refer to such curves as *boundary curves*. The set of all those interior and boundary curves that are each of length less than $d_{\max}$ together form the set $\mathcal{C}$.

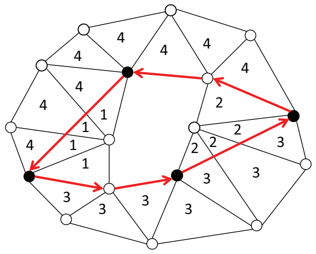
(b) Assemble valid loops. The next step in candidate patch generation is to assemble curves in $\mathcal{C}$ into closed loops. We store the connectivity information of the candidate curves using a graph $\mathcal{G} = (\mathcal{V}_s, \mathcal{C})$. We orient each curve such that it always starts from the vertex with the smaller index (i.e., the exact ordering is not important as long as it is consistent).

DEFINITION 1. *We define a loop l as a geometric realization of a simple cycle in graph $\mathcal{G}$ (i.e., one that never traverses the same edge twice). Assigning each loop l an initial orientation (as one of two possible orientations), we represent l using the following 1-chain form [Edelsbrunner and Harer 2010]:*

$$l := \sum_{c \in l} \sigma(c, l) \cdot c,$$

where $\sigma(c, l) = 1$ if the orientation of c agrees with that of l and $\sigma(c, l) = -1$ otherwise.

However, not all such loops are *valid* for our purpose. To define this notion, we introduce the following terminology. Each loop l may cut through faces of $\mathcal{S}$ so that l subdivides $\mathcal{S}$ into a new polygonal mesh $\mathcal{S}(l)$, which includes original (undivided) faces of $\mathcal{S}$ as well as faces that are original faces subdivided by edges of l (see inset). These new faces inherit their orientations from the orientations of the original faces in $\mathcal{S}$. Now for each polygonal face $f \in \mathcal{S}(l)$ that shares an edge with l, we say f is adjacent to the *left side* of l if the orientation of f, induced by the orientation of $\mathcal{S}$, agrees with the orientation of l at each point of intersection; if the orientations disagree at each point of intersection, we say f is adjacent to the *right side*. And if neither of these holds, we say that f is a bad face.



DEFINITION 2. *A loop l is* valid *if the following hold: (i) The loop is self-intersection free and simple; (ii) the loop separates its complement in $\mathcal{S}(l)$ into at least two disjoint components; and (iii) each component of $\mathcal{S}(l) \setminus l$ contains polygonal faces that are adjacent to* only one side *of l.*

(c) Perturb the loops. Since geodesic curves may cut through faces of $\mathcal{S}$ and thus do not necessarily consist of edges of the original mesh $\mathcal{S}$, it is inefficient to store and manipulate patches bounded by loops of the geodesic curves. We therefore perturb each of the valid loops generated above, so that they lie on edges of the original mesh. For any such geodesic curve $c = p_1 p_2 \cdots p_{m-1} p_m$, where each vertex p_i lies either at a mesh vertex or on a mesh edge, we create a perturbed curve $c' = p_1 q_2 \cdots q_{m-1} p_m$ as follows: First, all vertices of c that lie on mesh vertices are kept fixed. Else, for each vertex p_i that lies on a mesh edge $e := (v_1, v_2) \in \mathcal{E}$, we define q_i as the vertex v_1 or v_2 whichever is closer to p_i (see Figure 5). If p_i lies at the middle point between v_1 and v_2, then we always define

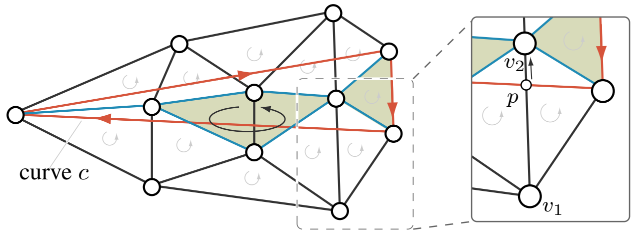


Fig. 5: *A triangular patch with its boundary (orange), perturbed boundary (blue), and the enclosed triangles (green). The perturbed boundary is oriented based on the orientation of the underlying surface.*

q_i as the vertex with the smaller index to break the tie. Note that the adjacent vertices of c' are either connected by a mesh edge or identical in which case we remove these identical vertices such that c' is given by a trace of mesh edges.

As the perturbed loop l' of a valid loop l may contain self-intersections, we use a region growing procedure to determine the left and right components of l'. Based on the initial orientation of l and the orientation of the underlying surface, region growing first initializes two sets of triangles $\mathcal{F}_L(l')$ and $\mathcal{F}_R(l')$ as the triangles that are adjacent to the left and right of the edges in l', respectively. Treating the mesh edges in the perturbed loop l' as a barrier, we then grow each triangle set separately using a breadth-first search. The resulting triangle sets $\mathcal{F}_L(l')$ and $\mathcal{F}_R(l')$ are called the left and right components of l', respectively.

PROPOSITION 1. *For each valid loop l', the left and right components $\mathcal{F}_L(l')$ and $\mathcal{F}_R(l')$ of its perturbed loop l' satisfy*

$$\mathcal{F}_L(l') \cap \mathcal{F}_R(l') = \emptyset, \quad \mathcal{F}_L(l') \cup \mathcal{F}_R(l') = \mathcal{F}.$$

Moreover, if both the two groups of components separated by l contain at least one entire face of the original mesh, then both $\mathcal{F}_L(l')$ and $\mathcal{F}_R(l')$ are non-empty.

PROOF. See Appendix A. □

(d) Select candidate patches. We are now ready to define the candidate patches that will be passed to the next stage of the NRG-subdivision extraction algorithm. For each perturbed loop l, where both sets $\mathcal{F}_L(l)$ and $\mathcal{F}_R(l)$ are non-empty, we call the set with the smaller total area the *inside* of l (or, an arbitrarily one, if the areas of the these two sets are identical.) We orient l such that its inside is always on its left side.

DEFINITION 3. *A* candidate patch *is the inside of a perturbed valid loop l with non-empty interior, together with l itself.*

Let a patch P be bounded by a perturbed valid loop l. Then in what follows, we will denote the inside of l by P° and the loop l itself as the boundary $\text{BOUND}(P)$ of P. Note that the boundary $\text{BOUND}(P)$ could be different from the topological boundary ∂P of P for instance when P° contains a hole (i.e., a component of the boundary of $\mathcal{S}$). Recall that such patches need not be topological discs.

DEFINITION 4. *For each curve c in the loop that makes up a patch P, we say c is on the* left-hand side *of P if $c \in \text{BOUND}(P)$ and $\sigma(c, \text{BOUND}(P)) = 1$. Similarly, c is on the* right-hand side *of P if $c \in \text{BOUND}(P)$ and $\sigma(c, \text{BOUND}(P)) = -1$.*

3.2 Geodesic Subdivision

We now introduce the notion of *valid patch collection*. In the abstract, a patch collection is very similar to a simplicial complex (c.f., [Edelsbrunner and Harer 2010]) except that the patches in a patch collection can be quadrilateral and not simply connected.

DEFINITION 5. *A* valid patch collection *$(\mathcal{M}_V, \mathcal{M}_C, \mathcal{M}_P)$ is given by a vertex set $\mathcal{M}_V$, a (shortest geodesic) curve set $\mathcal{M}_C$, and a set of valid patches $\mathcal{M}_P$ such that (i) the curve set $\mathcal{M}_C = \cup_{P \in \mathcal{M}_P} \text{BOUND}(P)$ comprises all the curves in the boundaries of patches in the patch set $\mathcal{M}_P$, and (ii) the vertex set $\mathcal{M}_V$ comprises all the end points of curves in the curve set $\mathcal{M}_C$.*

We denote the patch collection specified by all the candidate patches, the curves forming their boundaries, and the endpoints of these curves from Section 3.1 as $\overline{\mathcal{M}} := (\overline{\mathcal{M}_V}, \overline{\mathcal{M}_C}, \overline{\mathcal{M}_P})$. Note that $\overline{\mathcal{M}_C}$ is usually a subset of candidate curve set $\mathcal{C}$ since some candidate curves may not belong to any valid loop.

Note that in a patch collection, the patches may overlap with each other in patch interiors (see Figure 4). Now we are ready to define a geodesic subdivision.

DEFINITION 6. *We say a valid patch collection $\mathcal{M} = (\mathcal{M}_V, \mathcal{M}_C, \mathcal{M}_P)$ is a* geodesic subdivision *if for any pair of patches $P_i, P_j \in \mathcal{M}_P$, where $P_i \cap P_j \neq \emptyset$, then $P_i \cap P_j$ either (a) consists of a single point, which is an element of $\mathcal{M}_P$, or (b) consists of an entire curve c which is an element of $\mathcal{M}_C$ and which is an element of $\text{BOUND}(P_i)$ and $\text{BOUND}(P_j)$.*

DEFINITION 7. *We say a geodesic subdivision $\mathcal{M}$ is a* complete geodesic subdivision *if $\cup_{P \in \mathcal{M}_P} P = \mathcal{S}$. Otherwise, we say $\mathcal{M}$ is a* partial geodesic subdivision.

Equivalent constraints. In order to be able to formulate NRG-subdivision extraction as an integer program, we must re-phrase the definition of a geodesic subdivision in the form of equivalent constraints. To do so, we will need the following notions.

DEFINITION 8. *For each curve $c \in \mathcal{M}_C$, denote $\mathcal{A}(c, \mathcal{M}_P) = \{P | P \in \mathcal{M}_P, c \in \text{BOUND}(P)\}$. Let $\mathcal{A}_L(c, \mathcal{M}_P) \subset \mathcal{A}(c, \mathcal{M}_P)$ and $\mathcal{A}_R(c, \mathcal{M}_P) \subset \mathcal{A}(c, \mathcal{M}_P)$ be the patches on the left and right hand sides of curve c, respectively. The boundary of a patch collection $\mathcal{M}$ is then defined as $\text{BOUND}(\mathcal{M}) := \{c | c \in \mathcal{M}_C, |\mathcal{A}_L(c, \mathcal{M}_P)| \neq |\mathcal{A}_R(c, \mathcal{M}_P)|\}$.*

We now state in the following proposition a set of constraints such that if they are satisfied by a valid patch collection, then that patch collection forms a complete geodesic subdivision.

PROPOSITION 2. *A valid patch collection $\mathcal{M}$ is a complete geodesic subdivision of $\mathcal{S}$ if it satisfies the following constraints:*

— ***Curve constraint****: Each interior curve of $\mathcal{M}$ has the same number of patches on both its sides (left and right), i.e., for every $c \in \mathcal{M}_C$ with $c \not\subset \partial\mathcal{S}$, we have*

$$d_{\mathcal{M}}(c) := |\mathcal{A}_L(c, \mathcal{M}_P)| - |\mathcal{A}_R(c, \mathcal{M}_P)| = 0. \tag{1}$$

—***Uniqueness constraint****: The mesh face f_0 with the smallest index[1] is contained in exactly one patch of $\mathcal{M}$, i.e.,*

$$|\mathcal{M}_P(f_0)| = 1, \tag{2}$$

[1] Actually, one can replace f_0 by an arbitrary mesh face and the uniqueness constraint has the same effect. Essentially, the curve constraint propagates the uniqueness constraint.

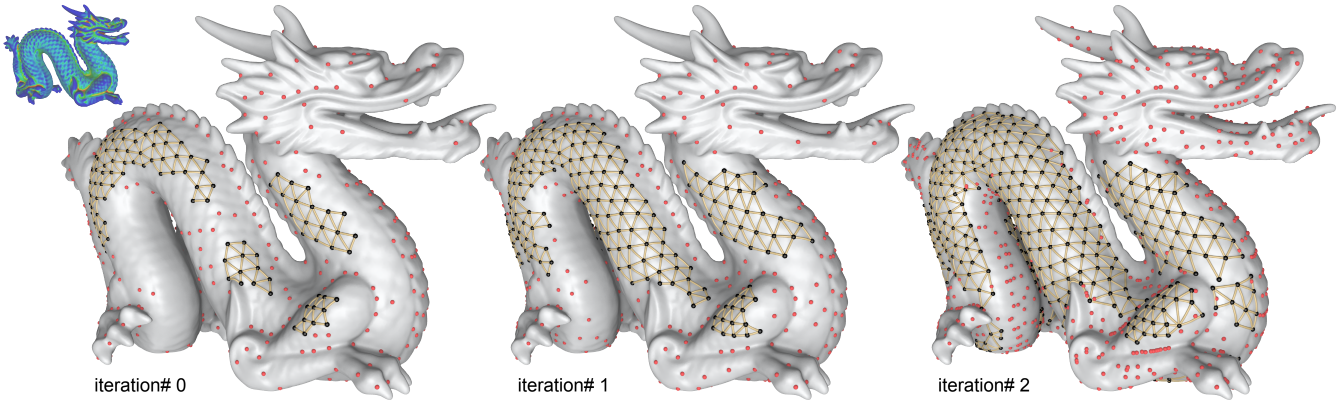


Fig. 6: *Extracted NRG-subdivision, shown in yellow, after multiple iterations. Samples are initially placed based on extremals of the heat kernel signature maps computed with small time scale (see top left). In each iteration, the active samples, $x_v = 1$, are marked in black, while the inactive, $x_v = 0$, ones are marked in red.*

where $\mathcal{M}_P(f) \subset \mathcal{M}_P$ denotes the set of patches that contain face f.

PROOF. See Appendix A. □

The constraints can be formulated as:

COROLLARY 1. *A complete geodesic subdivision $\mathcal{M}$ satisfies*

$$\forall c \in \mathcal{M}_C : \quad \begin{cases} |\mathcal{A}_L(c,\mathcal{M}_P)| = |\mathcal{A}_R(c,\mathcal{M}_P)| = 1, & c \not\subset \partial\mathcal{S} \\ |\mathcal{A}_L(c,\mathcal{M}_P)| + |\mathcal{A}_R(c,\mathcal{M}_P)| = 1, & \text{otherwise.} \end{cases}$$

For a partial geodesic subdivision $\mathcal{M}$, the curve constraints are invalid for curves in the BOUND(M). Further, as BOUND(M) is unknown, we cannot enforce the curve constraints exactly. Ignoring the curve constraints, a sufficient set of constraints for partial geodesic subdivision enforces the uniqueness constraints for all mesh faces of $\mathcal{S}$.

PROPOSITION 3. *A valid patch collection $\mathcal{M}$ is a partial geodesic subdivision of $\mathcal{S}$ if it satisfies the following uniqueness constraints:*

$$|\mathcal{M}_P(f)| \leq 1, \quad \forall f \in \mathcal{F}. \tag{3}$$

In practice, Equation 3 would incur a huge set of constraints, which makes the induced optimization problem hard to solve. As the curve constraints have the potential to propagate the uniqueness constraints, we instead enforce the curve constraints in a soft manner, i.e., minimizing the sum of $d_{\mathcal{M}}(c)$ over all the curves in $\mathcal{M}_C$. This strategy allows us to only consider a small subset of mesh faces for the uniqueness constraints and leads to an optimization problem of much smaller size, which works well in practice.

3.3 NRG-subdivision Extraction

We first present how to extract complete NRG-subdivision, then we show how to adopt the formulation for partial NRG-subdivision extraction. As the constraints for being a complete geodesic subdivision have been described in Equations 1 and 2, we proceed to introduce the objective function for evaluating the regularity score of patch collections.

Regularity energies. We introduce two energies for evaluating the regularity of any patch collection: the *patch regularity energy* and the *vertex regularity energy*.

The patch regularity energy is defined in terms of a regularity score w_P for each patch P that captures the local evidence of patch P being in the underlying NRG-subdivision. The precise expression of w_P is application-specific and will be given in the next section (see Figure 4 and Section 4). Once we have w_P, we define the patch regularity energy of a patch collection $\mathcal{M}$ as the sum of the negative regularity scores of the patches in $\mathcal{M}$:

$$R_p(\mathcal{M}) = -\sum_{P \in \mathcal{M}_P} w_P. \tag{4}$$

The vertex regularity energy is defined in terms of a regularity index $r_{\mathcal{M}}(v)$ for a vertex $v \in \mathcal{M}_V$, namely the deviation from the canonical valence η (where $\eta = 6$ for triangular patches and $\eta = 4$ for quad patches). In other words

$$\big||\mathcal{A}(v,\mathcal{M}_P)| - \eta\big|,$$

where $\mathcal{A}(v,\mathcal{M}_P)$ denotes the set of patches in $\mathcal{M}_P$ whose boundary loops contain vertex v in the graph. The vertex regularity energy of a patch collection $\mathcal{M}$ is then given as the cumulative vertex regularity:

$$R_v(\mathcal{M}) = \sum_{v \in \mathcal{M}_V} w_v \big||\mathcal{A}(v,\mathcal{M}_P)| - \eta\big|, \tag{5}$$

where w_v represents the confidence of the regularity of vertex v. For complete NRG-subdivision extraction, we set $w_v = 0$ if $v \in \partial\mathcal{S}$, and $w_v = 1$ otherwise.

Note that the patch and vertex regularity are not mutually independent as higher patch regularity drives each vertex to take the canonical valence η. In practice, however, we found that incorporating the vertex regularity term increases robustness.

Constrained optimization formulation. By combining Equations 1–5, we arrive at the following constrained optimization for complete NRG-subdivision extraction as minimization of $R_p(\mathcal{M}) + \lambda_v R_v(\mathcal{M})$ as:

$$\begin{aligned} \min_{\mathcal{M} \subset \overline{\mathcal{M}}} \quad & -\sum_{P \in \mathcal{M}_P} w_P + \lambda_v \sum_{v \in \mathcal{M}_V} w_v \big||\mathcal{A}(v,\mathcal{M}_P)| - \eta\big| \\ \text{so that,} \quad & |\mathcal{M}_P(f_0)| = 1, \\ & |\mathcal{A}_L(c,\mathcal{M}_P)| - |\mathcal{A}_R(c,\mathcal{M}_P)| = 0, \quad \forall c \in \mathcal{C}_I, \end{aligned} \tag{6}$$

where $\mathcal{C}_I = \{c | c \in \overline{\mathcal{M}}_C, c \not\subset \partial\mathcal{S}\}$. Note that the curve constraint should hold for all curves in the curve set $\mathcal{C}_I$ because $|\mathcal{A}_L(c, \mathcal{M}_P)| = |\mathcal{A}_R(c, \mathcal{M}_P)| = 0, \forall c \notin \mathcal{M}_C$. The parameter λ_v controls the importance of vertex regularity ($\lambda_v = 1$ in our tests).

Integer program formulation. The above optimization effectively amounts to extracting a suitable subset from patches $\overline{\mathcal{M}}_P$. We parameterize an arbitrary patch collection $\mathcal{M}$ by associating with each patch $P \in \overline{\mathcal{M}}_P$ a binary indicator z_P, where $z_P = 1$ if $P \in \overline{\mathcal{M}}_P$, and $z_P = 0$ otherwise.

In terms of the patch indicators, the patch regularity term defined in Equation 4 becomes

$$R_p(\mathcal{M}) = -\sum_{P \in \overline{\mathcal{M}}_P} w_P z_P. \qquad (7)$$

To formulate the vertex regularity term, we associate each sample $v \in \overline{\mathcal{M}}_V$ with a binary indicator x_v where $x_v = 1$ if $v \in \mathcal{M}_V$ and $x_v = 0$ otherwise. Combining both the patch indicators and vertex indicators, we rewrite the cumulative vertex regularity energy as,

$$R_v(\mathcal{M}) = \sum_{v \in \overline{\mathcal{M}}_V} w_v \left| \sum_{P \in \mathcal{A}(v, \overline{\mathcal{M}}_P)} z_P - \eta x_v \right|. \qquad (8)$$

Note that the vertex indicators are determined by the patch indicators, since $x_v = \max_{P \in \mathcal{A}(v, \overline{\mathcal{M}}_P)} z_P$. As the vertex regularity term pushes $x_v = 0$ if $z_P = 0 \quad \forall P \in \mathcal{A}(v, \overline{\mathcal{M}}_P)$, we relax these constraints as the following inequalities:

$$z_P \le x_v, \quad \forall P \in \overline{\mathcal{M}}_P, \quad \forall v \in \{v | P \in \mathcal{A}(v, \overline{\mathcal{M}}_P)\}. \qquad (9)$$

Using these patch indicators, we rewrite the constraints in Equations 1 and 2 respectively as,

$$\sum_{P \in \mathcal{A}_L(c)} z_P = \sum_{P \in \mathcal{A}_R(c)} z_P, \quad \forall c \in \mathcal{C}_I, \ \text{and} \quad \sum_{P \in \overline{\mathcal{M}}_P(f_0)} z_P = 1, \qquad (10)$$

where we have shortened $\mathcal{A}_L(c, \overline{\mathcal{M}}_P)$ and $\mathcal{A}_R(c, \overline{\mathcal{M}}_P)$ as $\mathcal{A}_L(c)$, $\mathcal{A}_R(c)$, respectively to simplify the notation.

Finally, combining Equations 7–10, we can convert the constrained optimization problem described in Equation 6 as solving the following integer program:

$$\begin{aligned} \min_{z_P, x_v \in \{0,1\}} \quad & -\sum_{P \in \overline{\mathcal{M}}_P} w_P z_P + \lambda_v \sum_{v \in \overline{\mathcal{M}}_V} w_v \Big| \sum_{P \in \mathcal{A}(v, \overline{\mathcal{M}}_P)} z_P - \eta x_v \Big| \\ \text{so that,} \quad & z_P \le x_v, \quad \forall P \in \overline{\mathcal{M}}_P, \quad \forall v \in \{v | P \in \mathcal{A}_P(v)\}, \\ & \sum_{P \in \mathcal{A}_L(c)} z_P = \sum_{P \in \mathcal{A}_R(c)} z_P, \quad \forall c \in \mathcal{C}_I, \\ & \sum_{P \in \overline{\mathcal{M}}_P(f_0)} z_P = 1. \end{aligned} \qquad (11)$$

Partial NRG-subdivision extraction. In the case of partial NRG-subdivision extraction, curves that lie on the boundary of the underlying NRG-subdivision no longer satisfy the curve constraint. Hence, we reformulate the curve constraint as an energy term

$$R_c(\mathcal{M}) = \sum_{c \in \mathcal{C}_I} w_c \left| \sum_{P \in \mathcal{A}_L(c)} z_P - \sum_{P \in \mathcal{A}_R(c)} z_P \right|, \qquad (12)$$

where w_c denotes our estimate on the possibility of curve c lying in interior of the underlying NRG-subdivision. We estimate w_c as

$$w_c = \min(w_c^L, w_c^R),$$

where $w_c^L = \max_{P \in \mathcal{A}_L(c)} w_P$ and $w_c^R = \max_{P \in \mathcal{A}_R(c)} w_P$. Intuitively, a curve c is likely to be in the interior of the underlying NRG-subdivision if there are regular patches on both sides of c resulting in higher w_c.

Further, the vertex regularity term should only count interior vertices of the underlying NRG-subdivision. Using the curve interior weight, we modify the vertex interior weights in Equation 5 as

$$w_v = \min_{c \in \mathcal{A}(v, \overline{\mathcal{M}}_C)} w_c,$$

where $\mathcal{A}(v, \overline{\mathcal{M}}_C)$ denotes the set of incident curves at v.

As the curve constraints have the potential to propagate the uniqueness constraints, we observed that it is sufficient to enforce the uniqueness constraints at a subset of mesh faces:

$$\sum_{P \in \mathcal{M}_P(f)} z_P \le 1, \quad \forall f \in \mathcal{F}^*, \qquad (13)$$

where $\mathcal{F}^* \subset \mathcal{F}$. Empirically, we found choosing $\mathcal{F}^*$ as a uniform sampling of $\mathcal{F}$ of size $0.05|\mathcal{F}|$ to be sufficient.

Combining Equations 7, 8, 12, and 13, we arrive at the following integer program formulation for partial NRG-subdivision extraction:

$$\begin{aligned} \min_{z_P, x_v \in \{0,1\}} \quad & -\sum_{P \in \overline{\mathcal{M}}_P} w_P z_P + \lambda_v \sum_{v \in \overline{\mathcal{M}}_V} w_v \Big| \sum_{P \in \mathcal{A}(v, \overline{\mathcal{M}}_P)} z_P - \eta x_v \Big| \\ & + \lambda_c \sum_{c \in \mathcal{C}_I} w_c \Big| \sum_{P \in \mathcal{A}_L(c)} z_P - \sum_{P \in \mathcal{A}_R(c)} z_P \Big| \\ \text{so that,} \quad & z_P \le x_v, \quad \forall P \in \overline{\mathcal{M}}_P, \quad \forall v \in \{v | P \in \mathcal{A}(v, \overline{\mathcal{M}}_P)\}, \\ & \sum_{P \in \mathcal{M}_P(f)} z_P \le 1, \quad \forall f \in \mathcal{F}^*, \end{aligned} \qquad (14)$$

where λ_c controls the importance of curve constraints ($\lambda_c = 0.5$ in our tests). The user can prescribe a preference for small but very regular partial NRG-subdivision, or larger but approximate partial NRG-subdivision by controlling the parameter λ_v.

3.4 Linear Programming Relaxation

We solve the integer programs formulated as Equations 11 and 14 using linear programming relaxation (LPR), wherein binary indica-

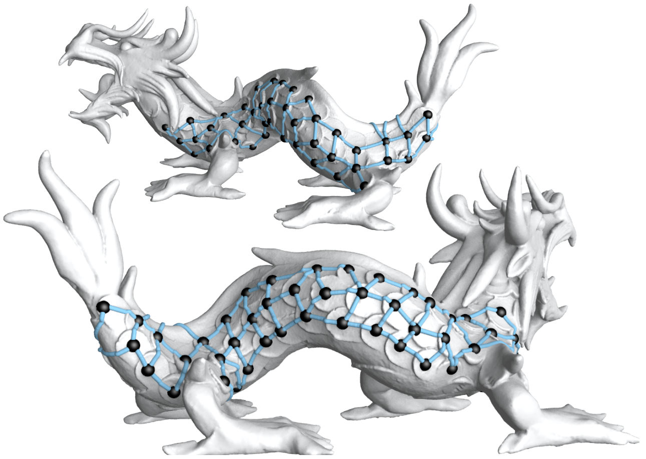


Fig. 7: *NRG-subdivision extracted on the dragon model. The extracted quad structure has a lower regularity score compared to the triangular near-regular mesh shown in Figure 2-left.*

tors are allowed to take real values in the interval $[0, 1]$. We prefer using linear relaxations as opposed to other integer program solvers due to the following reasons: (i) linear relaxations generate large-scale sparse linear programs that can be efficiently solved using interior-point methods [Boyd and Vandenberghe 2004; Grant and Boyd 2011]; (ii) typically require on the order of seconds to solve systems involving 50-100k variables; and (iii) the linear relaxations of our integer programs are usually tight.

Integrality conditions. In the Appendix, we present conditions when the LPRs return integer solutions, for both complete NRG-subdivision extraction (see Equation 17) and partial NRG-subdivision extraction. For complete NRG-subdivision extraction, the integrality condition implies that if the underlying NRG-subdivision $\mathcal{M}^*$ has the smallest patch regularity energy and regular interior vertices (or the vertex regularity is dropped), then solving the linear programming relaxation returns an integer solution that identifies this NRG-subdivision $\mathcal{M}^*$. On the other hand, if the vertex regularity and patch regularity adversely compete against each other, then the LPR of Equation 11 can yield fractional solutions, in which a rounding procedure has to be applied. Similarly, for partial NRG-subdivision extraction, the integrality condition means that the relaxation discovers the underlying partial NRG-subdivision if the vertex regularity, curve constraints and the patch regularity are mutually consistent.

In practice, for the examples shown in this paper, we found that the LRPs yield integer solutions. This is because the sample points come from feature extraction and candidate patches usually have high regularity scores. We also tested the LPRs on synthetic data sets, where we perturb the sample points from their ground-truth positions. Experimental results show that the LPRs still generate integer solutions if the perturbation is less than half of the sampling density (see Section 5).

Rounding strategy. In the presence of real-valued solutions, we use an iterative procedure to round them into integer solutions. The basic idea is to maintain two sets of patches, $\mathcal{F}_1$ and the remaining unclaimed patches. The set $\mathcal{F}_1$ includes all the patches whose indicators are fixed at 1, i.e., the currently selected patches. In each step, we identify the maximum indicator variable z^*, among the patches neighboring to the patches already in $\mathcal{F}_1$. We expand $\mathcal{F}_1$ to include the face corresponding to z^*. We then modify the linear relaxation to remove the indicator z^* and solve the updated LPR. Note that in the first step, if $\mathcal{F}_1$ is empty, z^* is extracted as the maximum among all the face indicators.

This rounding strategy does not, however, guarantee an optimal integer solution. Moreover, it is also possible that the reduced linear program becomes infeasible at some point, if the wrong patches are selected. In our tests we did not encounter such a situation. In general, an optimal integer solution can be found using branch-and-bound methods [Schrijver 1986]. But the worse case complexity of a branch-and-bound method is exponential.

4. APPLICATIONS

We show two applications of the sample-based NRG-subdivision extraction approach. First, we describe how to apply it to detect near-regular structures on 3D shapes, e.g., scales on dragon models. Second, we introduce an extension to establish markerless point based correspondences between 3D shapes.

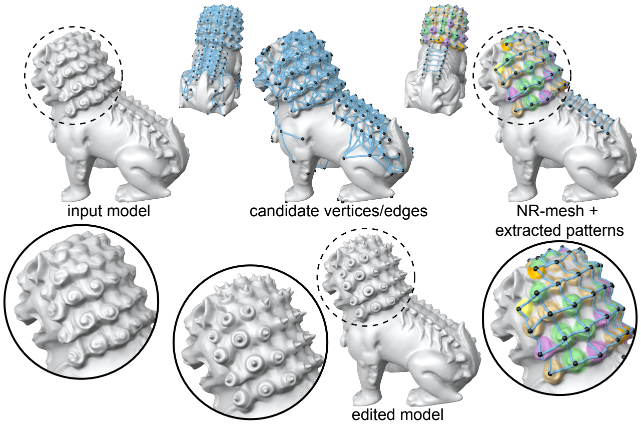


Fig. 8: *Detecting near-regular patterns on a surface. (Top) Input model; feature samples extracted using the local extrema of heat kernel signature maps, and candidate edges formed by connecting neighboring samples; extracted NR-mesh with detected base element. (Bottom) Edited model with a replaced base element.*

4.1 Near-Regular Structure Extraction

Recall that a near-regular structure on a surface $\mathcal{S}$ consists of a set of repeating elements or texels $\{\mathcal{X}_i\}$ and a NRG-subdivision $\mathcal{M}$ that captures of the positions of repeating elements. In this section, we formally define these repeating elements as texels, each of which represents a submesh of $\mathcal{S}$. To efficiently extract these texels, we assume that the geometric shapes of texels are approximately related to each other by similarity transformations (as is the case for all examples shown in this paper). Note that in contrast to Pauly et al. [2008], our NRG-subdivision are not required to manifest as regular grids in a transformation domain having 1-, 2-, or 3-parameter generators.

NRG-subdivision extraction. First, we extract the underlying NRG-subdivision $\mathcal{M}$ using the method presented in Section 3. For initial sample points $\mathcal{V}_s$, we use the local maxima of heat kernel signature (HKS) maps [Sun et al. 2009], i.e., $k_t(x, x), x \in \mathcal{S}$, where $k_t(\cdot, \cdot)$ denotes the heat-kernel operator (i.e., how one unit of heat diffuses across the surface), and t represents the scale of the HKS. Please refer to the original paper for computational details. In our implementation, we use a small value of $t = d_s/2$, where d_s is the averaged distance between each sample point to its closest neighbors. In this case, the corresponding HKS map is known to be related to isometry-invariant Gaussian curvatures [Sun et al. 2009]. In our tests, we set the maximum length of a shortest geodesic curve $d_{\max} = 3d_s$.

In the context of NRG-subdivision extraction, the regularity score w_P of each P is assigned as a combination of two factors: shape score $s_{\text{shape}}(P)$ and the edge consistency score $s_{\text{edge}}(P)$, i.e.,

$$w_P = s_{\text{shape}}(P) \cdot s_{\text{edge}}(P).$$

The shape score measures the deviation of patch P from a regular configuration, and is defined as $s_{\text{shape}}(P) = \exp(-a_P^2/2\sigma^2)$,where a_P is the maximum deviation of the angles of its vertices from the regular angle α ($\alpha = \pi/3$ for triangles and $\alpha = \pi/2$ for quads; $\sigma = \pi/4$ in all our examples). The angle at each vertex is defined as the angle between the two geodesic curves intersecting at this vertex. Edge consistency encodes the similarity of local geometry at the connected points, and is measured as the minimum consistency of its edges $s_{\text{edge}}(P) = \min_{c\in\text{BOUND}(P)}(1 - d(c))$, where $d(c)$ denotes the normalized heat kernel signature difference at the end points of curve c [Sun et al. 2009].

We solve for NRG-subdivision with triangular or quadrilateral patches. In most cases, however, the type of the pattern is unknown to start with. Therefore, we build two sets of candidate patches, one containing triangular patches and another containing quadrilateral patches. We run the detection procedure separately on both candidate sets and retain the NRG-subdivision with the higher regularity score (compare Figures 2 and 7). For NRG-subdivision with multiple connected components, i.e., patches in different components do not intersect, we select per connected component. In this case, we may have both triangular and quadrilateral patches on one model (see Figure 9).

In certain scenarios, only a handful of reliable features may be available, for example many of the scales on the Stanford-dragon are imperfect (see Figure 6) resulting in unstable local features. Starting with a dense set of sample points, it is computationally expensive to include all the potential features. Instead, we progressively extract the NRG-subdivision. Starting from a small set of salient, yet reliable, features we build a temporary NRG-subdivision. Here the saliency of a sample is defined as the averaged HKS difference to its neighboring vertices. Based on the learned structure, we prune sample points that are covered by patches of the learned structure and append the input set of sample points to include neighboring samples (whose geodesic distance to vertices of the learned structure is less than 2 times the averaged distance between adjacent vertices). We rerun the method on the enriched set of sample points to obtain a refined NRG-subdivision. Thus, we use the partial (global) patterns to effectively make use of even low strength local geometric features.

Texel extraction. We initialize the texels $\mathcal{X}_i$ as the surface Voronoi regions centered around the extracted points p_i, i.e., the 0-dimensional entities of the NRG-subdivision $\mathcal{M}$. We then use an alternating optimization procedure to refine the texels.

In each iteration, we first keep the texels and compute the relative similarity transformation T_i that aligns the shape of a base texel, say $\mathcal{X}_0$, with that of each other texel $\mathcal{X}_i$. To compute T_i, we first apply iterative closest point-to-plane (ICP) registration [Chen and Medioni 1992] to solve for the relative similarity transformation $\overline{T}_{i\to j}$ between each pair of neighboring texels $(\mathcal{X}_i, \mathcal{X}_j) \in \mathcal{N}_T$. Here we say two texels are neighbors if the points that capture their positions are connected by an edge in $\mathcal{M}$. Then, we compute T_i as,

$$\min_{\{T_i\}} \sum_{(\mathcal{X}_i,\mathcal{X}_j)\in\mathcal{N}_T} \|T_j - \overline{T}_{i\to j}T_i\|_F \qquad (15)$$

with T_0 being set to the identity transform and $\|.\|_F$ denoting Frobenius norm. We use the Gauss-Newton method described in [Krishnan et al. 2005] to solve Equation 15. Note that instead of restricting the transformations to lie on any predefined grid, using the data we recover an appropriate regular connecting structure connecting among the texel-pairs, possibly with missing elements.

We now use transformations T_i^{-1} to bring all the texels $\mathcal{X}_i$ to the local coordinate system of $\mathcal{X}_0$, and take the union of the transformed texels as the new base-texel, while keeping the correspondence fixed. We trim the base texel by removing boundary planar regions to retain only the characteristic geometry. Next, for each instance i, we transfer back the base texel using each T_i, apply ICP registration with the surface $\mathcal{S}$, and trim out a texel estimate. Individual texel centers p_i are computed using respective texel estimates, and their surface Voronoi regions used as the updated $\mathcal{X}_i$. We alternate between updating aligning transformations and texel recomputation. In practice, we found the transformations become steady after 3-5 iterations.

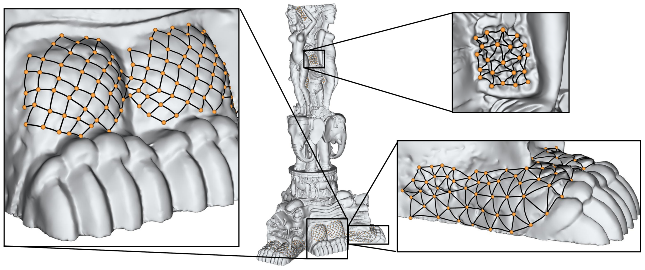


Fig. 9: *Our formulation handles detection of different NRG-subdivisions, quadrilateral and triangular patterns in this case, in a common framework.*

4.2 Markerless Correspondence Extraction

We adapt our framework for regularity detection to establish correspondences between two connected and oriented triangular meshes by extracting a canonical regular structure across the two shapes. The basic idea is to define the correspondence structure between two surfaces as a hyper NRG-subdivision, whose vertices, curves, and patches are vertex correspondences, curve correspondences, and patch correspondences, respectively.

Hyper-patch collection. We construct a hyper-patch collection $\overline{\mathcal{H}} = (\overline{\mathcal{H}}_V, \overline{\mathcal{H}}_C, \overline{\mathcal{H}}_P)$ between a pair of input surface meshes $\mathcal{S}$ and $\mathcal{T}$. We start by placing uniformly spaced samples on both shapes, $\mathcal{S}$ and $\mathcal{T}$, using HKS seeded farthest-point sampling to produce sample sets $\mathcal{V}_\mathcal{S}$ and $\mathcal{V}_\mathcal{T}$, respectively. We apply the procedure described in Section 3 to generate candidate curves and candidate triangular patches on both shapes.

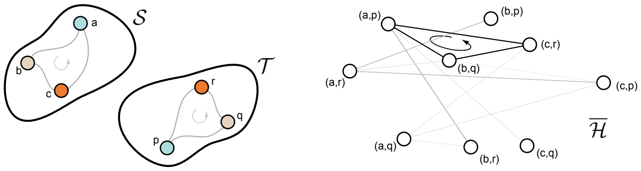


Fig. 11: *Correspondence detection between two oriented manifold meshes $\mathcal{S}, \mathcal{T}$ amounts to NRG-subdivision extraction on a hyper-patch collection $\overline{\mathcal{H}}$. We insert hyper-patch $(P, P') := ((a, p), (b, q), (c, r))$, if patches $P = (a, b, c)$ and $P' = (p, q, r)$ match the orientations of $\mathcal{S}$ and $\mathcal{T}$, respectively. Hyper-patch (P, P') is oriented based on orientations of $\mathcal{S}$ and $\mathcal{T}$.*

For each pair of sample points $v \in \mathcal{V}_\mathcal{S}$ and $v' \in \mathcal{V}_\mathcal{T}$, we create a hyper-vertex $(v, v') \in \overline{\mathcal{H}}_V$. Consistency of descriptor values, if they can be reliably computed on the respective meshes, is used to prune the hyper-vertex set. In this paper, we only connect each vertex to its 20-nearest neighbors on the other shape in terms of the HKS maps (see Section 4.1).

For each pair of hyper-vertices (v_1, v_1') and (v_2, v_2'), where there exist shortest geodesic curves $c_{12} = (v_1, v_2)$ and $c_{12}' = (v_1', v_2')$ connecting them, we introduce a hyper-curve $(c_{12} = (v_1, v_2), c_{12}' = (v_1', v_2')) \in \mathcal{H}_C$ if the geodesic distances are such that $d((c, c')) := |g_\mathcal{S}(v_1, v_2) - g_\mathcal{T}(v_1', v_2')|$ is within twice the average sampling spacing, with $g_\mathcal{X}(p, q)$ denoting the geodesic distance between points p and q on manifold $\mathcal{X}$. Note that the two

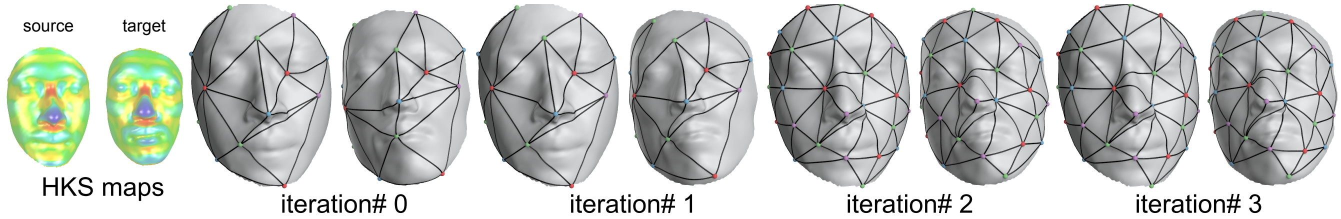


Fig. 10: *Markerless correspondence detection between a pair of faces. Our method alternates between topological and geometric optimization to produce a consistent mesh structure across the face pairs. This example is challenging due to lack of any dominating or locking features on the faces. We use HKS maps (left) to compute weights for evaluating candidate correspondence pairs (see Section 4.2).*

end "points" of this hyper-curve are hyper-vertices (v_1, v_1') and (v_2, v_2').

For each triplet $((v_1, v_1'), (v_2, v_2'), (v_3, v_3'))$, we check if either (i) $P = (v_1, v_2, v_3)$ and $P' = (v_1', v_2', v_3')$ construct [2] valid patches on $\mathcal{S}$ and $\mathcal{T}$, respectively, or (ii) $P = (v_1, v_3, v_2)$ and $P' = (v_1', v_3', v_2')$ construct valid patches on $\mathcal{S}$ and $\mathcal{T}$, respectively. If yes, we add the suitable triplet (P, P') to the hyper-patch set $\mathcal{H}_P$ with the associated orientation. Otherwise, we do not include the hyper-patch (see Figure 11). Note that the boundary $\text{BOUND}(P, P')$ of this hyper-patch is given by $((v_i, v_j), (v_i', v_j')), ij \in \{12, 23, 31\}$. The hyper-patches that are adjacent to the left and right sides of each hyper-curve are defined accordingly.

DEFINITION 9. *A hyper-patch collection $\mathcal{H} = (\mathcal{H}_V, \mathcal{H}_C, \mathcal{H}_P)$ is a hyper-geodesic subdivision if the intersection each pair of hyper-patches in $\mathcal{H}_P$ is (i) empty, or (ii) a hyper-curve in $\mathcal{H}_C$, or (iii) a hyper-vertex in $\mathcal{H}_V$.*

Based on the deviation of curve-wise geodesic distances we assign weight of hyper-patch (P, P') as

$$w_{(P,P')} = \exp(-\max_{(c,c')\in\text{BOUND}(P,P')} d((c, c'))^2/2\sigma^2),$$

where we set parameter $\sigma = \max_{(c,c')\in\mathcal{H}_C} d((c, c'))/2$ in our tests.

Hyper NRG-subdivision extraction. Global correspondence detection now amounts to compute a NRG-subdivision $\mathcal{H}$ from $\overline{\mathcal{H}}$. Accordingly, we modify the uniqueness constraint as

$$|\mathcal{H}_P(f_1)| = |\mathcal{H}_P(f_2)| = 1,$$

where f_1 and f_2 are arbitrary faces on $\mathcal{S}$ and $\mathcal{T}$, respectively. Here $\mathcal{H}_P(f_i) \subset \mathcal{H}_P$ denotes the hyper-patches that contain f_i on each surface. In the case of partial correspondence extraction, we modify the uniqueness constraints as

$$|\mathcal{H}_P(f)| \leq 1, \quad \forall f \in \mathcal{F}_\mathcal{S}^* \cup \mathcal{F}_\mathcal{T}^*,$$

where $\mathcal{F}_\mathcal{S}^*$ and $\mathcal{F}_\mathcal{T}^*$ are uniform face samples of $\mathcal{S}$ and $\mathcal{T}$, respectively. The corresponding integer programming formulations and linear programming relaxations are adapted accordingly. Note that for non-rigid shape matching, vertex regularity for NRG-subdivision extracted from $\overline{\mathcal{H}}$ is irrelevant, and hence the vertex regularity term is ignored.

Geometric optimization and re-sampling. The original samplings of the source and target meshes are performed independently, oblivious of the underlying correspondence. As a result, the extracted NRG-subdivision correspondence is typically imprecise due to sampling artifacts. We use the correspondence extracted in the topological optimization step to align the source shape with the target shape using non-rigid registration [Wand et al. 2007], and project the sample positions from the deformed source $\mathcal{S}$ to redefine sample locations on the undistorted target $\mathcal{T}$. Note that at this stage, only those points that are covered by the extracted NRG-subdivision from the source shape participate, affecting regions of the target sample distribution.

We treat the new positions as landmarks, and use farthest point sampling to add additional samples, typically 5% of the current sample count. As farthest point sampling is more accurate given reliable landmarks, the NRG-subdivision extracted typically grows across iterations. This process is terminated once the near-regular structure stops growing, typically in 2-4 iterations (see Figure 10).

5. EVALUATION AND DISCUSSION

We evaluated our approach on a variety of input models, both synthetic and real-world and of varying quality and complexity. Our framework is directly applicable for a large set of challenging problems. While for some of the presented problem instances specialized solutions exist, in other scenarios we are not aware of other successful attempts, e.g., extraction of patterns on the dragon models (see Figures 2 and 6), or automatic correspondence detection across meshes with different topology (see Figure 12).

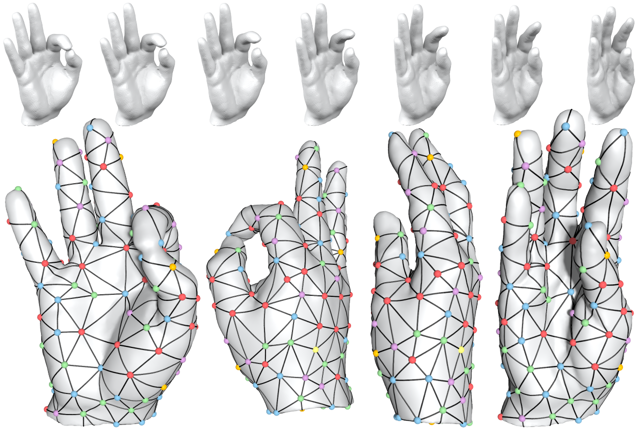


Fig. 12: *Automatic correspondence detection between two hand poses of varying topology. We simultaneously optimize over sample point placements and connectivity extraction to find a consistent topology across the two hand poses. The algorithm selects the open-hand connectivity as a persistent explanation for both the poses. (Top) We use the inferred correspondence to create shape space interpolation of the hand poses [Kilian et al. 2007].*

[2] We mean they are vertices of a patch in the clock-wise orientation.

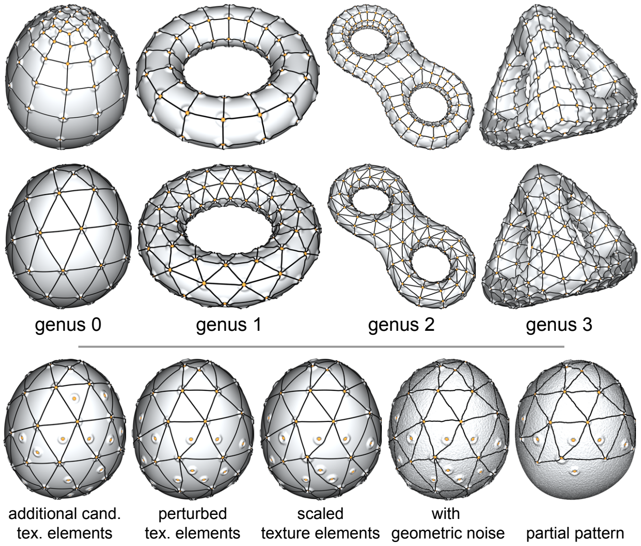


Fig. 13: *Detected NRG-subdivision on synthesized models with varying genus, perturbed textured elements, and in presence of geometric noise (see also Figure 14).*

NRG-subdivision extraction. We first tested the stability of our NRG-subdivision extraction procedure on a dataset with available ground truth (see Figure 13-top). The dataset consists of four models with genus ranging from 0 and 3. For each model, we generate both a triangular NRG-subdivision and a quad NRG-subdivision, using periodic global parameterization [Ray et al. 2006]. Given modified texture elements as input, we evaluate the performance of our method by comparing the detected NRG-subdivision with the corresponding known ones (i.e., ground-truth information).

We consider combinations of five modifications to the original set of texture elements as input test datasets (see Figure 13-bottom): (i) we add random texture elements; (ii) we perturb each texture element from its original position in a random direction by a random geodesic distance in $[0, \delta]$, where the displacement ratio δ controls the level of perturbation; (iii) we scale each texture element by a random variable in $[s_{min}, 1]$ to account for the variation in real patterns; (iv) we add geometric noise of size ϵ in the normal direction of the underlying model, and finally (v) we crop out a portion of texture elements to generate cases for partial NRG-subdivision detection. Note that the first two modifications simulate uncertainties arising from feature detection and/or sampling methods.

For each pattern, we used five different combination of modifications listed above instead of trying all possible combinations. In test (a) and test (b), we added 100% and 200% random texture elements, respectively. In test (c), we built on test (b) and scaled texture elements such that $s_{\min} = 0.5$. In test (d), we built on test (c) and added $\epsilon = 1\%$ noise to the underlying model. Finally, in test (e), we created a partial pattern from test (d) where 50% percent of the texture elements are removed. In each test scenario, we used δ from 0 to $0.5s$ where s is the sampling density of the underlying pattern. In our experiment, we made 20 different choices for δ and generated 20 patterns for each fixed set of modification parameters. We evaluate the performance of our method in terms of the detection ratio $(\#correct - \#incorrect)/\#groundtruth$. As the detected pattern might contain additional texture elements, we align the detected pattern with the original pattern by snapping each texture element to the closest original texture element. Figure 14 plots the detection ratio versus the displacement ratio δ.

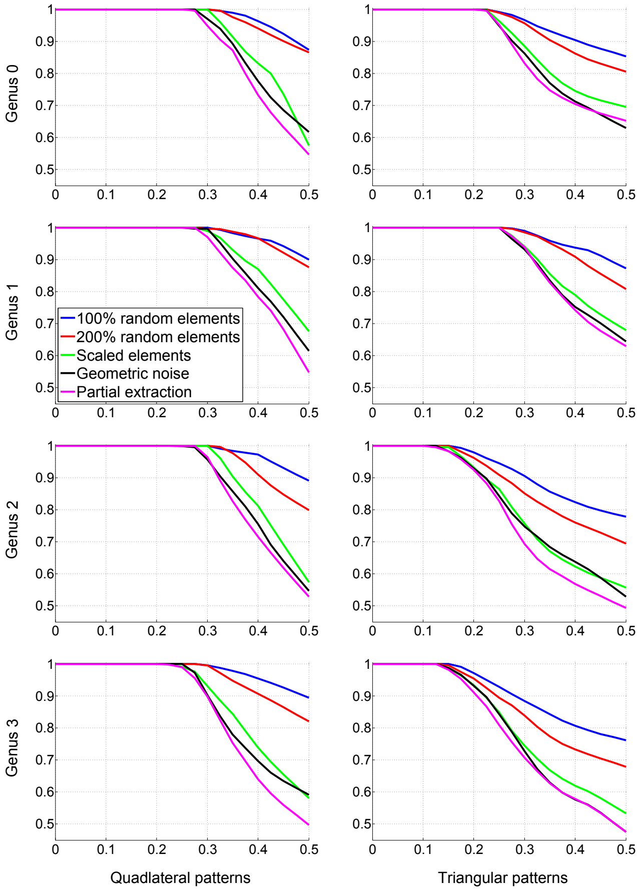


Fig. 14: *The pattern detection ratio versus the texture element displacement ratio on various scenarios. Statistics are collected on the near-regular patterns shown in Figure 13 (see text for more details).*

Our detection method does a perfect job when the texture elements are near-regular, e.g., $\delta < 0.2s$. However, the detection ratio deteriorates with increasing displacement ratio as the resultant placement of texture elements progressively becomes arbitrary. Overall, the detection ratio on the quadrilateral NRG-subdivision is higher than that of triangular NRG-subdivision, particularly when the displacement ratio is relatively small. This can be understood from the fact that regularity constraints on quadrilateral patches are stronger than that on triangular patches. In other words, when displacement ratio is small, it is easier to find incorrect regular triangular patches that include additional texture elements than for quad-pattern detection, since quad patches require coincidence of four texture elements.

NRG-subdivision editing. Extracting NRG-subdivision from surfaces greatly simplifies structure-preserving editing and manipulations, which are otherwise difficult to perform. In order to edit a NRG-subdivision, we first establish a dense correspondence between each texel $\mathcal{X}_i$ and the base texel $\mathcal{X}_0$. We use T_i to initially align texel $\mathcal{X}_i$ to the base texel $\mathcal{X}_0$, and apply non-rigid registration [Wand et al. 2007] to improve the alignment. For each point in

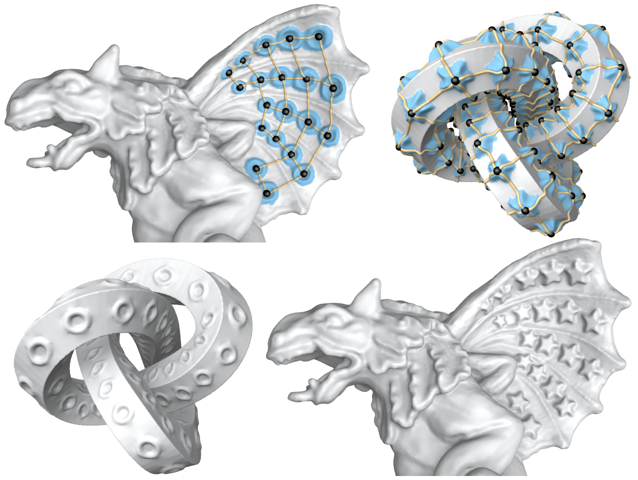


Fig. 15: *(Top) Detected NRG-subdivision on gargoyle and knot models with extracted patched highlighted in blue. (Bottom) Edited models obtained by mutually swapping the detected base texels, while retaining original repetition structures with corresponding deformations.*

texel $\mathcal{X}_i$, we take its closest point on the aligned copy of $\mathcal{X}_0$ as its corresponding point.

We now describe two structure-preserving manipulations. (i) We modify the base texel $\mathcal{X}_0$, while preserving the original NRG-subdivision and use Laplacian surface editing [Sorkine et al. 2004] to transfer the edits back onto each texel using the extracted T_i. Figures 2 and 8 show typical example modifications where we replace the scales and the spiral hair patterns, respectively on the two dragon models with a circular hair pattern. Similarly, Figure 15 shows the result of swapping the base texels between the detected NR-structures extracted from the two models. In this example, the correspondences between the two base texels are computed by mapping each of them onto a canonical disc [Floater 2003]. (ii) We also reduce the number of repetitions while preserving the repetition structure to shrink the dragon (see Figure 2-top).

Markerless correspondence extraction. We extensively tested our global correspondence detection method on three benchmark data sets, namely SCAPE [Anguelov et al. 2005], SHREC07 [Giorgi et al. 2007], and TOSCA [Bronstein et al. 2008]. We compare our correspondence detection results with available ground truth in Table I, and refer to the supplementary materials for results on all the 190 models. Overall, the performance of our method is comparable to or slightly better than Mobius voting [Lipman and Funkhouser 2009; Kim et al. 2010] in terms of both geodesic error of correspondences and the averaged percentage of model areas that are in correspondence.

First, we tested our approach on the TOSCA high-resolution benchmark, which consists of 80 objects classified into 7 classes, with objects in the same class having a common triangulation, thus providing a ground truth for comparison. For each object class, we select the first object and match it against all the others in this class. Note that consistent correspondences are obtained by fixing the sample points on the selected object. As objects in the same class contain similar sets of local features, our method finds very high quality correspondences. In a more challenging test, we matched 5 different animal models (see Figure 16).

Next, we tested on the SCAPE data set, which contains meshes of a single person in a wide range of poses. Our method is still able to find consistent correspondences between the template and all example poses (see Figure 2 and supplementary material). In case of large deformations between the template model and the example poses, in some cases we observe slippage across featureless regions, e.g., elbow regions.

	Cat	Dog	Wolf	Horse	Centaur	Gorillas	Female	Male	SCAPE
p_m	96.4	95.4	99.6	94.7	93.6	95.6	94.9	95.6	92.1
e_{geo}	1.5	1.3	0.7	1.4	1.6	1.3	1.1	0.9	2.1

Table I. : *Statistics on different object classes from the TOSCA and SCAPE benchmarks. Here, p_m represents the averaged percentage of model areas that are in correspondence, and e_{geo} represents averaged geodesic error of correspondences with respect to percentage of the averaged bounding boxes of available ground truth.*

Finally, we have tested our method on the human and four legged classes from the SHREC07 watertight dataset, which is particularly interesting because objects differ significantly even in the same semantic class. This challenges the sampling stage of our approach. We obtained high quality correspondences even in this case. In our experiments, the method typically converges in two iterations. In the second pass, the correspondences serves as landmarks for additional samples that subsequently help recover correspondence across thin parts (see Figure 17).

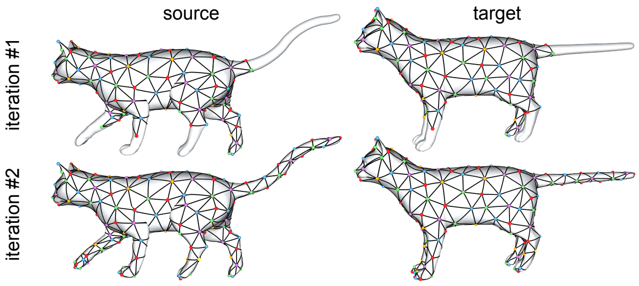


Fig. 17: *Global correspondence detection involving models with thin features can require multiple iterations, with initial partial correspondence helping to adaptively sample for subsequent iterations.*

Performance. Since we only connect neighboring points, the size of the patches is linear in the number of input points. For sparse linear programs, linear programming converges in $O(\sqrt{n})$ time; each time we solve a sparse linear system it takes $O(n^{3/2})$, with n being the number of input samples. We use the CVX package [Grant and Boyd 2011]. In practice, solving the LPRs using interior point method requires about 30-50 iterations. In Table II, we list the running times for the various examples.

Limitations. Our algorithm is based on an integer programming formulation and naturally inherits some of its limitations. Since we solve for linear programming relaxations, handling very large point sets, involving 100k-s of variables, can become intractable. Such scenarios can occur for detection of higher order relations, e.g., global correspondence.

Additionally, in absence of distinctive or characteristic features, the sample points can slide along the mesh surface without affecting the regularity score, for example, near the elbow of the human models, or other thin structures (see supplementary material). In absence of additional information, such slippage of nodes or associated base patches is unavoidable (see also [Mitra et al. 2010]).

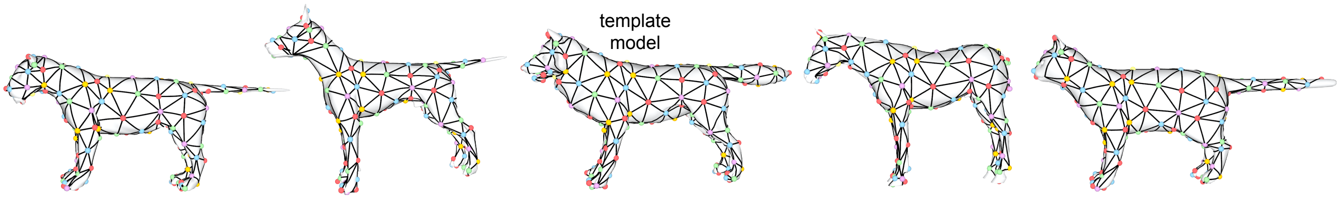


Fig. 16: *Inter-class global correspondence detection across five animals in the TOSCA dataset (see supp. for complete results).*

Finally, our framework cannot directly handle scanned point clouds as we require an initial oriented manifold surface to orient the candidate patches created for NRG-subdivision extraction.

6. CONCLUSIONS

We presented linear programming formulations for solving complete and partial near-regular structure detection. Our method optimizes for both topological connectivity and geometric placement to efficiently extract regular structures. We introduced integrality conditions for the associated linear programs, and empirically justified the effectiveness of the geometric relaxation, which is based on heuristic measures. We evaluated the robustness of our method on synthetic test cases and large collections of benchmark datasets, in presence of geometric and topological noise. We presented applications to NRG-subdivision detection on surfaces, structure-preserving manipulations, and markerless correspondence extraction.

Our framework for structure detection on 3D geometry and also on abstract simplicial complexes promises several interesting research avenues:

—Our method works with any computable definition of regularity of patches and vertices. This naturally allows users to encode application specific regularity measures, e.g., for constrained and fabrication-aware meshing and structure-aware shape processing (see [Mitra et al. 2013]).

—The ability to automatically establish global correspondences across large collections of object collections holds the promise to learn variations in shape families and intuitively navigate the resultant shape spaces.

	$\lvert\overline{\mathcal{M}}_V\rvert$	$\lvert\overline{\mathcal{M}}_C\rvert$	$\lvert\overline{\mathcal{M}}_P\rvert$	n^*_P	t_{pre}	t_{lp}	n	t
knot	280	1320	1720	160	74.2s	15.1s	1	89.4s
gargoyle	916	2164	1530	23	232.1s	14.1s	1	246.2s
chin-drag.	108	348	272	49	151.2s	0.9s	1	152.1s
xyz-drag.	327	3482	1918	114	232.1s	21.6s	1	253.7s
Stan-drag.	1742	6276	5480	1335	121.4s	55.1s	3	529.5s
xyz-statue.	1265	5346	3480	855	321.4s	76.1s	2	794.5s
	$\lvert\mathcal{V}_{\mathcal{K}}\rvert$	$\lvert\mathcal{E}_{\mathcal{K}}\rvert$	$\lvert\mathcal{F}_{\mathcal{K}}\rvert$					
face	311	1378	785	70	92.2s	7.1s	4	90.2s
hand	382	4875	3927	284	145.6s	45.2s	4	764.4s
TOSCA	314	2871	3582	382	140.9s	75.3s	2	448.8s
SCAPE	360	2521	3897	392	191.1s	92.1s	2	586.4s
SHREC07	486	6571	5582	350	150.9s	280.3s	2	882.8s

Table II. : *Timing with 2.4GHz Intel CPU with 6GB RAM; n^*_P: number of patches/patch-correspondences in the extracted NRG-subdivision, t_{pre}: preprocessing time including descriptor computation and constructing simplicial complex, t_{lp}: averaged time for solving linear program at each iteration, n: number of total geometry optimization iterations, and t: total time.*

APPENDIX

A. PROPOSITION PROOFS

A.1 Sketch proof of proposition 1

The basic idea of the proof is to consider a continuous family of loops that interpolates the original loop and the perturbed loop, and study the change of the components that are separated by these loops. Denote the original loop as $l = p_1 \cdots p_n p_1$. Let $l' = p'_1 \cdots p'_n p'_1$ be the perturbed loop where $p'_i \in \mathcal{V}$ is the perturbed vertex of p_i. (Duplicated vertices are not removed.) Denote $l(t) = q_1(t) \cdots q_n(t) q_1(t),\ 0 \leq t \leq 1$ as the continuous family of loops that linearly interpolate between l and l', where $q_i(t) = (1-t)p_i + tp'_i$. It is easy to see that $l(t)$ lies on $\mathcal{S}$.

Lemma 1. *If l is valid, then for any $t \in [0, 1)$, $l(t)$ is also valid.*

Proof. It is easy to see that the order of the vertices of $l(t)$ on each edge is the same as that of $l(0) = l$. It follows that $l(t)$ is self-intersection free, and the topological structure of the mesh obtained by cutting $\mathcal{S}$ by $l(t)$ is the same as that of the mesh obtained by cutting $\mathcal{S}$ along l, which ends the proof. □

Let $\mathit{Left}(l(t))$ and $\mathit{Right}(l(t))$ be the left and right components of $l(t)$ in the mesh $\mathcal{S}(l(t))$, respectively. Denote $\mathbf{c}(f)$ as the barycenter of each face $f \in \mathcal{F}$. Let

$$\hat{\mathcal{F}}_L(l(t)) = \{f | f \in \mathcal{F}, \mathbf{c}(f) \in \mathit{Left}(l(t))\}$$
$$\hat{\mathcal{F}}_R(l(t)) = \{f | f \in \mathcal{F}, \mathbf{c}(f) \in \mathit{Right}(l(t))\}$$

be the set of triangles whose centers belong to $\mathit{Left}(l(t))$ and $\mathit{Right}(l(t))$, respectively. It is easy to see that for $t \geq 3/4$, $\hat{\mathcal{F}}_L(l(t))$ and $\hat{\mathcal{F}}_R(l(t))$ are fixed, and

$$\hat{\mathcal{F}}_L(l(t)) \cup \hat{\mathcal{F}}_R(l(t)) = \mathcal{F}, \quad \hat{\mathcal{F}}_L(l(t)) \cap \hat{\mathcal{F}}_R(l(t)) = \emptyset.$$

It remains to prove:

Lemma 2. *Let $t \geq 3/4$ be a constant, $\hat{\mathcal{F}}_L(l(t)) = \mathcal{F}_L(l(1))$ and $\hat{\mathcal{F}}_R(l(t)) = \mathcal{F}_R(l(1))$, where $\mathcal{F}_L(l(1))$ and $\mathcal{F}(l(1))$ are given by the region-growing procedure described in Section 3.*

Proof. First, we have $\hat{\mathcal{F}}_L(l(t)) \subset \mathcal{F}_L(l(1))$. This is because different connected components (where adjacent faces share edges) of $\hat{\mathcal{F}}_L(l(t))$contains faces that are adjacent to the left side of $l(1)$, and faces in the same connected component can be reached from each other using the region-growing process. It remains to prove $\hat{\mathcal{F}}_R(l(t)) \cap \mathcal{F}_L(l(1)) = \emptyset$. Suppose $\exists f \in \hat{\mathcal{F}}_R(l(t)) \cap \mathcal{F}_L(l(1))$. Then f is connected via adjacent faces to a face that is adjacent to the left of $l(1)$, which also belongs to $\hat{\mathcal{F}}_L(l(t))$. This means $f \in \hat{\mathcal{F}}_R(l(t)) \cap \hat{\mathcal{F}}_L(l(t))$, which ends the proof. □

A.2 Sketch proof of proposition 2

The basic idea is to cut the original mesh $\mathcal{S}$ into a new mesh $\mathcal{S}'$ using the boundary loops of patches in $\mathcal{M}_P$, and then prove that the curve constraints imply that the covering number of each face $f \in \mathcal{S}'$, i.e., the number of patches in $\mathcal{M}_P$ that f belongs to, is fixed for adjacent faces.

As we will see later, the proof requires that patches in M_P are non-degenerate, i.e., their boundaries are free of self-intersections. Similar to the proof of proposition 1, we thus consider patches formed by loops that are not fully perturbed. More precisely, for each patch $P \in \mathcal{M}_P$, consider a family of components $P(t), 0 \leq t \leq 1$ on S defined by the family of loops that interpolates $\textsc{bound}(P)$ and the unperturbed loop of $\textsc{bound}(P)$. Note that the curve constraints on components in $\mathcal{M}_P(t) := \cup_{P \in \mathcal{M}_P} P(t)$ are inherited from $\mathcal{M}_P$. As for $t > 3/4$, $P(t)$ and P cover the same set of barycenters of faces in S, it remains to prove that each face barycenter appears in $\mathcal{M}_P(t)$ a constant number of times[3].

Let $\mathcal{S}'$ be the triangle mesh generated by cutting the original mesh $\mathcal{S}$ using all boundary loops of $P(t) \in \mathcal{M}_P(t)$. For each face $f \in \mathcal{S}'$, we define its covering multiplicity cover(f) as

$$\text{cover}(f) = \sum_{P(t) \in \mathcal{M}_P(t)} \text{Id}(f \in P(t)),$$

where $\text{Id}(f \in P(t)) = 1$ if and only if $f \in P(t)$.

Given two adjacent faces f and f' in S' that share an edge e, we have

$$\text{cover}(f) - \text{cover}(f') = \sum_{f \in P(t), f' \notin P(t)} 1 - \sum_{f' \in P(t), f \notin P(t)} 1.$$

Consider the curves $l(t)$ (in boundary loops of $\mathcal{M}_P(t)$) that contain e. As components $P(t)$ are non-degenerate, it is easy to see that due to the curve constraints, $\sum\limits_{f \in P(t), f' \notin P(t)} 1$ and $\sum\limits_{f' \in P(t), f \notin P(t)} 1$ are equal to the total number of components that are adjacent to the left and right of these curves (the order does not matter). In other words, we have $\text{cover}(f) = \text{cover}(f')$. As $\mathcal{S}'$ is connected, cover(f) is a constant, which ends the proof.

B. INTEGRALITY CONDITIONS

Let us start with understanding the solution space of the linear relaxation of Equation 11.

DEFINITION 10. *We say a patch collection*[4] $\mathcal{M}$ *is valid if*

$$|\mathcal{A}_L(c, \mathcal{M})| = |\mathcal{A}_R(c, \mathcal{M})|, \quad \forall c \in \mathcal{C}_I \cap \mathcal{M}$$

for complete NRG-subdivision extraction and

$$max(|\mathcal{A}_L(c, \mathcal{M})|, |\mathcal{A}_R(c, \mathcal{M})|) = 1, \quad \forall c \in \mathcal{M}$$

for partial NRG-subdivision extraction.

LEMMA 3. *Let* $\mathcal{U}$ *denote the set of valid patch collections. For each* $\mathcal{M} \in \mathcal{U}$, *denote* $\mathbf{y}_{\mathcal{M}} = (\mathbf{x}_{\mathcal{M}}; \mathbf{z}_{\mathcal{M}}) \in [0,1]^{|\overline{\mathcal{M}}_P| + |\overline{\mathcal{M}}_V|}$ *where* $\mathbf{x}_{\mathcal{M}}$ *and* $\mathbf{z}_{\mathcal{M}}$ *collect vertex indicators and patch indicators of* $\mathcal{M}$, *respectively. For each* $v \in \mathcal{M}_V$, *let* $\mathbf{y}_v = (\mathbf{e}_v; \mathbf{0}) \in [0,1]^{|\overline{\mathcal{M}}_P| + |\overline{\mathcal{M}}_V|}$ *where* $\mathbf{e}_v$ *denotes the standard basis. Suppose* $\mathbf{x}$ *and* $\mathbf{z}$ *form a feasible solution to the linear relaxation of Equation 11 or Equation 14. Then* $\mathbf{y} = (\mathbf{z}; \mathbf{x})$ *can be decomposed as*

$$\mathbf{y} = \sum_{\mathcal{M} \in \mathcal{U}^+} t_{\mathcal{M}} \mathbf{y}_{\mathcal{M}} + \sum_{v \in \overline{\mathcal{M}}_V} t_v \mathbf{y}_v, \tag{16}$$

where $\mathcal{U}^+ \subset \mathcal{U}$ *is a subset of valid patches,* $t_{\mathcal{M}}$ *are positive and* t_v *are non-negative.*

PROOF. The proof is as follows: $\mathbf{y}$ can be represented as a linear combination of basic feasible solutions, which are given by either valid patch collections or individual vertices. We refer to [Schrijver 1986] for details about the properties of linear programs. □

Complete NRG-subdivision Extraction. The following theorem presents an integrity condition for complete NRG-subdivision extraction setting.

THEOREM 1. *A solution to the linear relaxation of Equation 11, which is described by a valid patch collection* $\mathcal{M}^* \in \mathcal{U}$, *is the unique optimal solution if*

$$\frac{R_p(\mathcal{M}^*) + \lambda_v E(\mathcal{M}^*)}{m(\mathcal{M}^*)} < \frac{R_p(\mathcal{M}) + \lambda_v E_{\mathcal{M}^*}(\mathcal{M})}{m(\mathcal{M})}, \forall \mathcal{M} \in \mathcal{U} \backslash \{\mathcal{M}^*\}, \tag{17}$$

where $m(\mathcal{M})$ *denotes the number of times that* $\mathcal{M}$ *covers the underlying surface* $\mathcal{S}$ *and*

$$E(\mathcal{M}^*) = R_v(\mathcal{M}^*) + \eta m(\mathcal{M}^*) \sum_{v \in \mathcal{S}^+} w_v,$$

$$E_{\mathcal{M}^*}(\mathcal{M}) = \sum_{v \in \mathcal{S}^+ \cap \mathcal{M}} (\eta + r_{\mathcal{M}}(v)) w_v - \sum_{v \in \mathcal{S}^- \cap \mathcal{M}} r_{\mathcal{M}}(v) w_v, \tag{18}$$

where $\mathcal{S}^+(\mathcal{S}^-) = \{v | v \in \mathcal{M}^*, r_{\mathcal{M}^*}(v) > (<) 0\}$ *denotes the set of positive(negative) singular vertices of the patch collection* $\mathcal{M}^*$.

PROOF. Suppose instead the optimal solution is given by

$$\overline{\mathbf{y}} = \sum_{\mathcal{M} \in \overline{\mathcal{U}}^+} \overline{t}_{\mathcal{M}} \mathbf{y}_{\mathcal{M}} + \sum_{v \in \overline{\mathcal{M}}_V} \overline{t}_v \mathbf{y}_v.$$

Let us consider the restricted linear program where $\mathcal{U}^+$ in the decomposition of the patch indicator vector $\mathbf{y}$ (See Equation 16) is a subset of $\overline{\mathcal{U}} = \overline{\mathcal{U}}^+ \cup \{\mathcal{M}^*\}$. Note that both $\mathbf{y}_{\mathcal{M}^*}$ and $\overline{\mathbf{y}}$ are feasible solutions of this restricted linear program.

Substituting Equation 16 into Equation 11, we can reformulate the restricted linear program by treating $t_{\mathcal{M}}$ and t_v as variables:

$$\begin{aligned}
\min & \sum_{\mathcal{M} \in \overline{\mathcal{U}}} R_p(\mathcal{M}) t_{\mathcal{M}} + \lambda_v \sum_{v \in \overline{\mathcal{M}}_V} |\sum_{\mathcal{M} \in \overline{\mathcal{U}}} r_{\mathcal{M}}(v) t_{\mathcal{M}} - \eta t_v| \\
\text{s.t.} & \sum_{\mathcal{M} \in \overline{\mathcal{U}}} m(\mathcal{M}) t_{\mathcal{M}} = 1, \\
& \sum_{\mathcal{M} \in \overline{\mathcal{U}}, v \in \mathcal{M}} t_{\mathcal{M}} + t_v \leq 1, \quad \forall v \in \overline{\mathcal{M}}_V \\
& 0 \leq t_{\mathcal{M}}, t_v, \quad \forall \mathcal{M} \in \overline{\mathcal{U}}, v \in \overline{\mathcal{M}}_V.
\end{aligned} \tag{19}$$

It remains to prove that $\mathbf{t}^*$, which is given by

$$t^*_{\mathcal{M}} = \left\{ \begin{array}{cc} 1/m(\mathcal{M}^*) & \mathcal{M} = \mathcal{M}^* \\ 0 & \text{otherwise} \end{array} \right., \quad t_v = 0, \quad \forall v \in \overline{\mathcal{M}}_V,$$

is the optimal solution to Equation 19. In this case, $\mathbf{y}$ cannot be the optimal solution since $\mathbf{t}^*$ is a better solution.

[3] If necessary, we can always slightly perturb the barycenter of a triangle such that it does not lie on any boundary of $P(t)$.

[4] Note that each patch can repeat multiple times in this case.

We complete the proof by contradiction. In fact, we only need to prove $\mathbf{t}^*$ is a strict local minimum since Equation 19 is a convex program and every strict local minimum of a convex optimization problem is also the global minimum [Boyd and Vandenberghe 2004]. Suppose there exists a better feasible solution $\mathbf{t} \neq \mathbf{t}^*$ where

$$\|\mathbf{t} - \mathbf{t}^*\|_\infty < 1/(\max_{v \in \mathcal{V}_s} \sum_{\mathcal{M} \in \overline{\mathcal{U}}} |r_{\mathcal{M}}(v)| + \eta).$$

First of all, it is easy to see that

$$\sum_{\mathcal{M} \in \overline{\mathcal{U}}} r_{\mathcal{M}}(v) t_{\mathcal{M}} - \eta t_v \geq 0 (\leq 0), \quad \forall v \in \mathcal{S}^+ (\mathcal{S}^-).$$

Thus, the objective value of Equation 19

$$\begin{aligned}
\geq \quad & \sum_{\mathcal{M} \in \overline{\mathcal{U}}} R_p(\mathcal{M}) t_{\mathcal{M}} + \lambda_v \sum_{v \in \mathcal{S}^+ \cup \mathcal{S}^-} w_v |\sum_{\mathcal{M} \in \overline{\mathcal{U}}} r_{\mathcal{M}}(v) t_{\mathcal{M}} - \eta t_v| \\
\geq \quad & \sum_{\mathcal{M} \in \overline{\mathcal{U}}} R_p(\mathcal{M}) t_{\mathcal{M}} - \lambda_v \sum_{v \in \mathcal{S}^-} w_v \sum_{\mathcal{M} \in \overline{\mathcal{U}}} r_{\mathcal{M}}(v) t_{\mathcal{M}} \\
+ \quad & \lambda_v \sum_{v \in \mathcal{S}^+} w_v (\sum_{\mathcal{M} \in \overline{\mathcal{U}}} r_{\mathcal{M}}(v) t_{\mathcal{M}} - \eta t_v) \\
\geq \quad & \sum_{\mathcal{M} \in \overline{\mathcal{U}}} R_p(\mathcal{M}) t_{\mathcal{M}} - \lambda_v \sum_{v \in \mathcal{S}^-} w_v \sum_{\mathcal{M} \in \overline{\mathcal{U}}} r_{\mathcal{M}}(v) t_{\mathcal{M}} \\
+ \quad & \lambda_v \sum_{v \in \mathcal{S}^+} w_v (\sum_{\mathcal{M} \in \overline{\mathcal{U}}} r_{\mathcal{M}}(v) t_{\mathcal{M}} - \eta (1 - \sum_{\mathcal{M}, v \in \mathcal{M}} t_{\mathcal{M}})) \\
= \quad & \sum_{\mathcal{M} \in \overline{\mathcal{U}}} t_{\mathcal{M}} (R_p(\mathcal{M}) + \lambda_v E_{\mathcal{M}^*}(\mathcal{M})) - \eta \lambda \sum_{v \in \mathcal{S}^+} w_v \\
> \quad & (R_p(\mathcal{M}^*) + \lambda_v E(\mathcal{M}^*)) / m(\mathcal{M}^*) - \eta \lambda_v \sum_{v \in \mathcal{S}^+} w_v \\
= \quad & \sum_{\mathcal{M} \in \overline{\mathcal{U}}} R_p(\mathcal{M}) t^*_{\mathcal{M}} + \lambda_v \sum_{v} w_v |\sum_{\mathcal{M} \in \overline{\mathcal{U}}} r_{\mathcal{M}}(v) t^*_{\mathcal{M}} - \eta t^*_v|,
\end{aligned}$$

which is a contradiction. □

COROLLARY 2. *Suppose there exists a valid patch collection $\mathcal{M}^*$ such that $\lambda_v R_v(\mathcal{M}^*) = 0$, e.g. either the vertex regularity term is not omitted or the vertex regularity of $\mathcal{M}^*$ is zero. Then $\mathcal{M}^*$ is the unique optimal solution to the linear relaxation of Equation 11 if*

$$R_p(\mathcal{M}^*) / m(\mathcal{M}^*) < R_p(\mathcal{M}) / m(\mathcal{M}), \forall \mathcal{M} \in \mathcal{U} \setminus \{\mathcal{M}^*\}.$$

Theorem 1 and Corollary 2 indicate that the linear relaxation of Equation 11 tends to return integer indicators if the vertex regularity term is consistent with the patch regularity term. On the other hand, the linear relaxation could return real indicators if these two terms adversely compete with each other.

Partial NRG-subdivision Extraction. The following theorem presents an integrality condition for partial NRG-subdivision extraction.

THEOREM 2. *Suppose $\sigma(\cdot) : \mathcal{F}^0 \rightarrow \mathbb{R}^+$ denotes an arbitrary positive function on face samples $\mathcal{F}^0$. Then a valid patch collection $\mathcal{M}^*$ is the unique optimal solution to the linear relaxation of Equation 14 if*

$$R_p(\mathcal{M}^*) + E'(\mathcal{M}^*) < \frac{R_p(\mathcal{M}) + E'_{\mathcal{M}^*}(\mathcal{M})}{\gamma_{\mathcal{M}^*}(\mathcal{M})}, \forall \mathcal{M} \in \mathcal{U} \setminus \{\mathcal{M}^*\} \qquad (20)$$

where,

$$\begin{aligned}
E'(\mathcal{M}^*) &= \lambda_v E(\mathcal{M}^*) + \lambda_c R_c(\mathcal{M}^*), \\
E'_{\mathcal{M}^*}(\mathcal{M}) &= \lambda_v E_{\mathcal{M}^*}(\mathcal{M}) + \lambda_c \sum_{c \in \mathcal{M}^*} w_c |d_{\mathcal{M}^*}(c)| d_{\mathcal{M}}(c)
\end{aligned}$$

and

$$\gamma_{\mathcal{M}^*}(\mathcal{M}) = \sum_{f \in \mathcal{F}^0_{\mathcal{M}^* \cap \mathcal{M}}} \sigma(f) / \sum_{f \in \mathcal{F}^0_{\mathcal{M}^*}} \sigma(f),$$

where $\mathcal{F}^0_{\mathcal{M}}$ denotes the set of face samples that are covered by $\mathcal{M}$.

PROOF. The proof is about similar to the proof of Theorem 1. Since $\mathcal{U}$ is finite in the partial NRG-subdivision extraction setting, we directly write down the equivalent linear programming formulation to Equation 14 as

$$\begin{aligned}
\min & \sum_{\mathcal{M} \in \mathcal{U}} R_p(\mathcal{M}) t_{\mathcal{M}} + \lambda_v \sum_{v \in \overline{\mathcal{M}}_V} w_v |\sum_{\mathcal{M} \in \mathcal{U}} r_{\mathcal{M}}(v) t_{\mathcal{M}} - \eta t_v| \\
& + \lambda_c \sum_{c \in \mathcal{C}_I} w_c |\sum_{\mathcal{M} \in \mathcal{U}} d_{\mathcal{M}}(c) t_{\mathcal{M}}| \\
\text{s.t.} & \sum_{\mathcal{M} \in \mathcal{U}_f} t_{\mathcal{M}} \leq 1, \quad \forall f \in \mathcal{F}^0, \\
& \sum_{\mathcal{M} \in \mathcal{U}, v \in \mathcal{M}} t_{\mathcal{M}} + t_v \leq 1, \quad \forall v \in \overline{\mathcal{M}}_V \\
& 0 \leq t_{\mathcal{M}}, t_v, \quad \forall \mathcal{M} \in \mathcal{U}, v \in \overline{\mathcal{M}}_V,
\end{aligned} \qquad (21)$$

where $\mathcal{U}_f$ denotes the set of proper patch collections that cover face f. Second, we have

$$\begin{aligned}
& \sum_{\mathcal{M} \in \mathcal{U}} \gamma_{\mathcal{M}^*}(\mathcal{M}) t_{\mathcal{M}} \\
= \quad & \sum_{\mathcal{M} \in \mathcal{U}} \sum_{f \in \mathcal{F}^0_{\mathcal{M}^* \cap \mathcal{M}}} t_{\mathcal{M}} \sigma(f) / \sum_{f \in \mathcal{F}^0_{\mathcal{M}^*}} \sigma(f) \\
= \quad & \sum_{f \in \mathcal{F}^0_{\mathcal{M}^*}} \sigma(f) (\sum_{\mathcal{M} \in \mathcal{U}_f} t_{\mathcal{M}}) / \sum_{f \in \mathcal{F}^0_{\mathcal{M}^*}} \sigma(f) \\
\leq \quad & 1.
\end{aligned}$$

Finally, for any feasible solution $\mathbf{t} \neq \mathbf{t}^*$ where

$$\|\mathbf{t} - \mathbf{t}^*\|_\infty \leq 1 / \max(\max_{c \in \mathcal{C}_I} \sum_{\mathcal{M} \in \mathcal{U}} |d_{\mathcal{M}}(c)|, \max_{v \in \overline{\mathcal{M}}_V} \sum_{\mathcal{M} \in \mathcal{U}} |r_{\mathcal{M}}(v)|),$$

we find that the objective value of Equation 21

$$\begin{aligned}
\geq \quad & \sum_{\mathcal{M} \in \mathcal{U}} R_p(\mathcal{M}) t_{\mathcal{M}} - \lambda_c \sum_{c, d_{\mathcal{M}^*}(c) < 0} w_c \sum_{\mathcal{M}} d_{\mathcal{M}}(c) t_{\mathcal{M}} \\
+ \quad & \lambda_c \sum_{c, d_{\mathcal{M}^*}(c) > 0} w_c \sum_{\mathcal{M}} d_{\mathcal{M}}(c) t_{\mathcal{M}} - \lambda_v \sum_{v \in \mathcal{S}^-} w_v \sum_{\mathcal{M} \in \mathcal{U}} r_{\mathcal{M}}(v) t_{\mathcal{M}} \\
+ \quad & \lambda \sum_{v \in \mathcal{S}^+} w_v (\sum_{\mathcal{M} \in \mathcal{U}} r_{\mathcal{M}}(v) t_{\mathcal{M}} - \eta (1 - \sum_{\mathcal{M}, v \in \mathcal{M}} t_{\mathcal{M}})) \\
= \quad & \sum_{\mathcal{M} \in \mathcal{U}} t_{\mathcal{M}} (R_p(\mathcal{M}) + E'_{\mathcal{M}^*}(\mathcal{M})) - \eta \lambda_v \sum_{v \in \mathcal{S}^+} w_v \\
> \quad & \sum_{\mathcal{M} \in \mathcal{U}} \gamma_{\mathcal{M}^*}(\mathcal{M}) t_{\mathcal{M}} (R_p(\mathcal{M}^*) + E'(\mathcal{M}^*)) - \eta \lambda_v \sum_{v \in \mathcal{S}^+} w_v \\
\geq \quad & (R_p(\mathcal{M}^*) + \mu R_c(\mathcal{M}^*) + \lambda R_v(\mathcal{M}^*)),
\end{aligned}$$

which ends the proof. □

ACKNOWLEDGMENTS

We would like to acknowledge the anonymous reviewers for their helpful comments and detailed suggestions, and Henry Adams, Adrian Butscher, Mirela Ben-Chen, Maks Ovsjanikov, and Helmut Pottmann for the extremely useful feedback and discussions. This work was supported by NSF grants CCF 1011228, Marie Curie Career Integration Grant 303541, ERC Starting Grant SmartGeometry 335373, a KAUST-Stanford AEA grant, a KAUST visiting scholarship, a Google research award, and a Stanford Graduate Fellowship.

REFERENCES

ANGUELOV, D., SRINIVASAN, P., KOLLER, D., THRUN, S., RODGERS, J., AND DAVIS, J. 2005. Scape: shape completion and animation of people. In *ACM SIGGRAPH 2005 Papers*. 408–416.

BERNER, A., WAND, M., MITRA, N. J., MEWES, D., AND SEIDEL, H.-P. 2011. Shape analysis with subspace symmetries. *Comput. Graph. Forum 30,* 2, 277–286.

BOKELOH, M., WAND, M., AND SEIDEL, H.-P. 2010. A connection between partial symmetry and inverse procedural modeling. In *ACM SIGGRAPH 2010 papers*. 104:1–104:10.

BOYD, S. AND VANDENBERGHE, L. 2004. *Convex Optimization*. Cambridge University Press, New York, NY, USA.

BRONSTEIN, A., BRONSTEIN, M., AND KIMMEL, R. 2008. *Numerical Geometry of Non-Rigid Shapes*, 1 ed. Springer Publishing Company, Incorporated.

CHEN, Y. AND MEDIONI, G. 1992. Object modelling by registration of multiple range images. *Image Vision Comput. 10*, 145–155.

EDELSBRUNNER, H. AND HARER, J. 2010. Computational topology: an introduction.

FLOATER, M. S. 2003. Mean value coordinates. *Comput. Aided Geom. Des. 20,* 1 (Mar.), 19–27.

GAL, R., SORKINE, O., MITRA, N. J., AND COHEN-OR, D. 2009. iwires: an analyze-and-edit approach to shape manipulation. In *ACM SIGGRAPH 2009 papers*. 33:1–33:10.

GIORGI, D., BIASOTTI, S., AND PARABOSCHI, L. 2007. Shape retrieval contest 2007: Watertight models track.

GRANT, M. AND BOYD, S. 2011. CVX: Matlab software for disciplined convex programming. `http://www.stanford.edu/~boyd/cvx/`.

HAYS, J., LEORDEANU, M., EFROS, A. A., AND LIU, Y. 2006. Discovering texture regularity as a higher-order correspondence problem. In *ECCV 2006, Proceedings, Part II*. 522–535.

KILIAN, M., MITRA, N. J., AND POTTMANN, H. 2007. Geometric modeling in shape space. In *ACM SIGGRAPH 2007 papers*. 64:1–64:9.

KIM, V. G., LIPMAN, Y., CHEN, X., AND FUNKHOUSER, T. A. 2010. Möbius transformations for global intrinsic symmetry analysis. *Comput. Graph. Forum 29,* 5, 1689–1700.

KRISHNAN, S., LEE, P. Y., MOORE, J. B., AND VENKATASUBRAMANIAN, S. 2005. Global registration of multiple 3d point sets via optimization-on-a-manifold. In *Proceedings of SGP'05*.

LI, M., LANGBEIN, F. C., AND MARTIN, R. R. 2006. Constructing regularity feature trees for solid models. In *Proc. Geometric Modeling and Processing; LNCS*. Springer, 267–286.

LIPMAN, Y. AND FUNKHOUSER, T. 2009. Mobius voting for surface correspondence. In *ACM SIGGRAPH 2009 papers*. 72:1–72:12.

LIU, S., MARTIN, R. R., LANGBEIN, F. C., AND ROSIN, P. L. 2007. Segmenting periodic reliefs on triangle meshes. In *IMA Conference on the Mathematics of Surfaces*. 290–306.

LIU, Y., LIN, W.-C., AND HAYS, J. 2004. Near-regular texture analysis and manipulation. In *ACM SIGGRAPH 2004 Papers*. 368–376.

MITRA, N. J., BRONSTEIN, A. M., AND BRONSTEIN, M. M. 2010. Intrinsic regularity detection in 3d geometry. In *ECCV 2010, Proceedings, Part III*. 398–410.

MITRA, N. J., GUIBAS, L. J., AND PAULY, M. 2006. Partial and approximate symmetry detection for 3d geometry. In *ACM SIGGRAPH 2006 Papers*. 560–568.

MITRA, N. J., PAULY, M., WAND, M., AND CEYLAN, D. 2012. Symmetry in 3d geometry: Extraction and applications. In *EUROGRAPHICS State-of-the-art Report*.

MITRA, N. J., WAND, M., ZHANG, H., COHEN-OR, D., AND BOKELOH, M. 2013. Structure-aware shape processing. In *EUROGRAPHICS State-of-the-art Report*.

OVSJANIKOV, M., MÉRIGOT, Q., MÉMOLI, F., AND GUIBAS, L. J. 2010. One Point Isometric Matching with the Heat Kernel. *Computer Graphics Forum 29,* 5 (July), 1555–1564. Proc. SGP 2010.

OVSJANIKOV, M., SUN, J., AND GUIBAS, L. J. 2008. Global intrinsic symmetries of shapes. *Comput. Graph. Forum 27,* 5, 1341–1348.

PARK, M., BROCKLEHURST, K., COLLINS, R. T., AND LIU, Y. 2009. Deformed lattice detection in real-world images using mean-shift belief propagation. *IEEE Trans. Pattern Anal. Mach. Intell. 31*, 1804–1816.

PAULY, M., MITRA, N. J., WALLNER, J., POTTMANN, H., AND GUIBAS, L. J. 2008. Discovering structural regularity in 3d geometry. In *ACM SIGGRAPH 2008 papers*. 43:1–43:11.

PODOLAK, J., SHILANE, P., GOLOVINSKIY, A., RUSINKIEWICZ, S., AND FUNKHOUSER, T. 2006. A planar-reflective symmetry transform for 3d shapes. In *ACM SIGGRAPH 2006 Papers*. 549–559.

RAY, N., LI, W. C., LÉVY, B., SHEFFER, A., AND ALLIEZ, P. 2006. Periodic global parameterization. *ACM Trans. Graph. 25*, 1460–1485.

SCHAFFALITZKY, F. AND ZISSERMAN, A. 1999. Geometric grouping of repeated elements within images. In *Shape, Contour and Grouping in Computer Vision*. Springer-Verlag, London, UK, 165–181.

SCHRIJVER, A. 1986. *Theory of linear and integer programming*. John Wiley & Sons, Inc., New York, NY, USA.

SORKINE, O., COHEN-OR, D., LIPMAN, Y., ALEXA, M., RÖSSL, C., AND SEIDEL, H.-P. 2004. Laplacian surface editing. In *Proceedings of the SGP'04*. 175–184.

SUN, J., OVSJANIKOV, M., AND GUIBAS, L. 2009. A concise and provably informative multi-scale signature based on heat diffusion. In *Proceedings of SGP '09*. 1383–1392.

TEVS, A., BOKELOH, M., WAND, M., SCHILLING, A., AND SEIDEL, H.-P. 2009. Isometric registration of ambiguous and partial data. In *Proceedings of CVPR 2009*. 1185–1192.

THOMPSON, D. W. 1945. *On Growth and Form*. Cambridge University Press.

TUYTELAARS, T., TURINA, A., AND VAN GOOL, L. 2003. Noncombinatorial detection of regular repetitions under perspective skew. *IEEE Trans. Pattern Anal. Mach. Intell. 25*, 418–432.

VAN KAICK, O., ZHANG, H., HAMARNEH, G., AND COHEN-OR, D. 2011. A survey on shape correspondence. *Comput. Graph. Forum 30,* 6, 1681–1707.

WAND, M., JENKE, P., HUANG, Q., BOKELOH, M., GUIBAS, L., AND SCHILLING, A. 2007. Reconstruction of deforming geometry from time-varying point clouds. In *Proceedings of SGP'07*. 49–58.

XIN, S.-Q. AND WANG, G.-J. 2009. Improving chen and han's algorithm on the discrete geodesic problem. *ACM Trans. Graphics 28*, 104:1–104:8.

XU, K., ZHANG, H., TAGLIASACCHI, A., LIU, L., LI, G., MENG, M., AND XIONG, Y. 2009. Partial intrinsic reflectional symmetry of 3d shapes. In *ACM SIGGRAPH Asia 2009 papers*. 138:1–138:10.